INTERACTIVE COMPUTER ASSISTED PART PROGRAMMING FOR CNC LATHE

A Thesis submitted
in partial fulfillment of the Requirements
for the Degree of

MASTER OF TECHNOLOGY

By

AMITABH PARASHAR

to the

DEPARTMENT OF INDUSTRIAL AND MANAGEMENT ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY

KANPUR

February, 2002

CERTIFICATE

This is to certify that the work contained in this thesis entitled ***Interactive Computer assisted Part Program generation For CNC Lathe*** by Amitabh Parashar has been carried out under our supervision and that this work has not been submitted elsewhere for the award of a degree.

Dr. Kripa Shanker
Professor and Head
Industrial & Management Engg.
Indian Institute of Technology
Kanpur

Dr. V.K.Jain
Professor
Deptt. of Mechanical Engg.
Indian Institute of Technology
Kanpur

February, 2002

Abstract

The present dissertation deals with the development of interactive part program generation software for CNC lathe using AutoCAD and Visual Basic. The graphic user interface has been developed using Visual Basic language. The user will make the drawing in AutoCAD and the software will read all the coordinates of the drawing automatically. After that the software will recognize the features in the drawing and then it will do the process sequencing for the initial stock removal. Lastly it will generate the CNC Part Program which can be read and fed to the CNC Machine. For other special operations like chamfering, threading, drilling etc. parametric programming approach has been attempted in which the user has to input the appropriate coordinates interactively as asked by the software. The package runs on IBM compatible Pentium 1 and onwards under the window environment.

ACKNOWLEDGEMENTS

I express deep gratitude to my guides Dr. Kripa Shanker and Dr. V. K. Jain for their valuable suggestions and encouragement. They were always available in spite of busy schedule. I will always be grateful for them for the interest they have shown in my research.

I learnt quite a bit from the courses I did as a part of my Master of Technology curriculum and I thank entire IME faculty for making learning a wonderful experience.

I am indebted to all my IME friends.

I thank to the staff of Central library, IME department, Hall-IV and CC, for all the help they had extended during my stay at IITK.

I thank all my friends in Hall-IV for making my stay memorable. I am always grateful to all my family members for the love and support they had given me during the tough periods.

I thank the whole manufacturing laboratory of Mechanical Department for their cooperation.

Contents

Chapter 1

Introduction

Chapter 4

Implementation and Results

Chapter 5

Conclusions and scope for future work

List of figures

List of tables

Chapter 1

Introduction

1.1. CNC machines: At a glance

Numerical control machining has been developed out of the need for higher productivity, lower cost and more precise manufacturing. This is the latest machine tool control system since the industrial revolution and can be considered as the most sophisticated form of automation for controlling machine tool, equipments and processes. There is hardly a facet of manufacturing that is not in some way touched by what these innovative machine tools can do. Everyone involved in the manufacturing environment is well aware of what is possible with these sophisticated machine tools. The design engineer, for example, must possess enough knowledge of CNC to perfect dimensioning and tolerancing techniques for workpieces to be machined on CNC machines. The tool engineer must understand CNC in order to design fixtures and cutting tools for use with CNC machines. Quality control people should understand the CNC machine tools used within their company in order to plan quality control and statistical process control accordingly. Production control personnel should be abreast of their company's CNC technology in order to make realistic production schedules. Managers, foremen, and team leaders should understand CNC well enough to communicate intelligently with fellow workers. And, it goes without saying that CNC programmers, setup people, operators, and others working directly with the CNC equipment must have an extremely good understanding of CNC. [Reference:1]

In the NC system, operation instructions are inputted to the machine as numbers and alphabets which are suitably coded for storing information. These instructions are then automatically carried out in the machine tool in predetermined sequence with preset or self adjusted speed, feed etc. without human intervention. Avoidance of human intervention, ommission of conventional tooling and fixturing and quick-change capablity of NC machine system are the primary factors considered to decide the level of acceptance of the NC machine tools for a particular job. Other maintainable advantages identified of NC machine tools over conventional machine with automation are:

1. Optimaization of cutting tool life and quality of jobs.

2. Possibility of making parts which are impossible in conventional machining systems.
3 Quick and more accurate inspection and detection of error in design and fabrication.
4 The improved automation, fewer mistakes caused by human error, consistency and predictable machining time for each workpiece.
5. Another benefit offered by most forms of CNC machine tools is flexibility. Since these machines are run from programs, running a different workpiece is almost as easy as loading a different program. Once a program has been verified and executed for one production run, it can be easily recalled the next time the workpiece is to be run This leads to yet another benefit, fast change-overs.

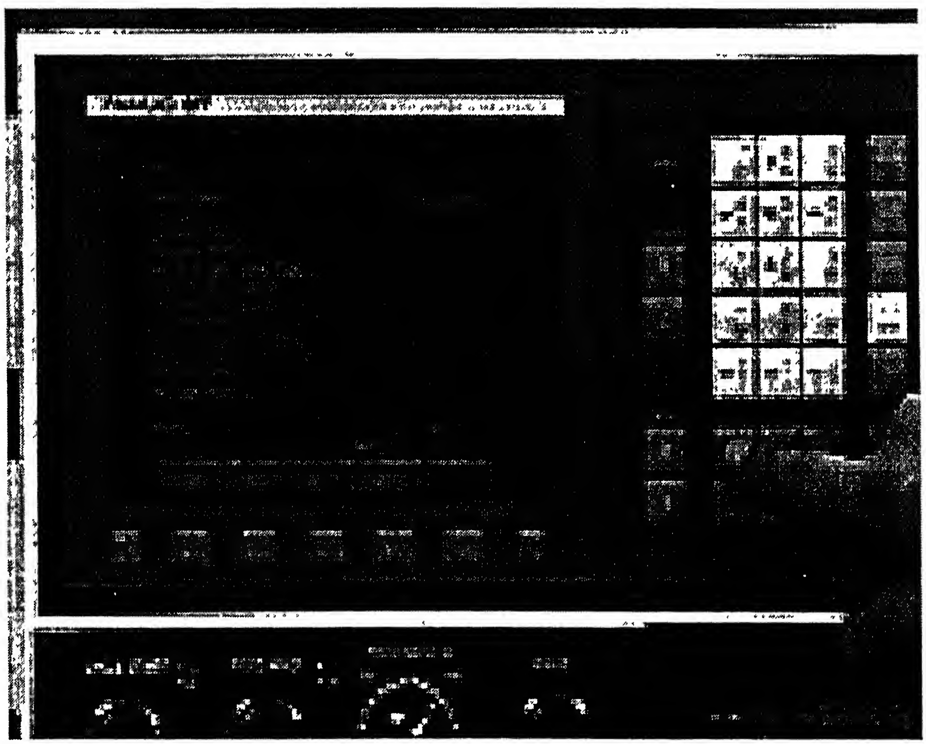


Figure 1.1. CNC control penal, [Reference 21]

In the original NC systems the physical components are hard wired i e. the circuits and components can perform their respective functions only and are not flexible to adopt changes. In CNC system the physical components are software units In soft wired units the loaded program in computer makes the control unit operate to suit the need of machinist The

Machine Control Unit, the heart of the NC system underwent a great development with the introduction of very large scale integrated circuits. The new features not available earlier are:

- Provisions of incremental and absolute programming which are incorporated by G90 (absolute) and G91(incremental) codes.
- Provision of inch or metric data input through G70 (inch) and G71 (metric).

The most basic function of any CNC machine is automatic, precise, and consistent motion control. Rather than applying completely mechanical devices to cause motion as is required on most conventional machine tools, CNC machines allow motion control in a revolutionary manner. All forms of CNC equipment have two or more directions of motion, called axes. These axes can be precisely and automatically positioned along their lengths of travel. The two most common axis types are linear (driven along a straight path) and rotary (driven along a circular path). Instead of causing motion by turning cranks and handwheels as is required on conventional machine tools, CNC machines allow motions to be commanded through programmed commands. Generally speaking, the motion type (rapid, linear, and circular), the axes to move, the amount of motion and the motion rate (feedrate) are programmable with almost all CNC machine tools. Accurate positioning is accomplished by the operator counting the number of revolutions made on the handwheel plus the graduations on the dial. The drive motor is rotated a corresponding amount, which in turn drives the ball screw, causing linear motion of the axis. A feedback device confirms that the proper amount of ball screw revolutions have occurred. A CNC command executed within the control (commonly through a program) tells the drive motor to rotate a precise number of times. The rotation of the drive motor in turn rotates the ball screw and the ball screw causes drives the linear axis. A feedback device at the opposite end of the ball screw allows the control to confirm that the commanded number of rotations has taken place. [Reference: 21]

1.2. Three basic motions in CNC machine

Three motion types are available on almost all forms of CNC equipment. An example program is shown that stresses the use of all three. These motion types share two things in common. First, they are all modal. This means they remain in effect until changed. If for example, several motions of the same kind are to be given consecutively, the corresponding G code need only be specified in the first command. Second, the ENDPOINT of the motion is specified in each motion command. The current position of the machine will be taken as the starting point.

1.2.1. Rapid motion (also called positioning)

This motion type (as the name implies) is used to command motion at the machine's fastest possible rate. It is used to minimize non-productive time during the machining cycle. Common uses for rapid motion include positioning the tool to and from cutting positions, moving to clear clamps and other obstructions, and in general, any non-cutting motion during the program.

Usually this rate is extremely fast (some machines boast rapid rates of well over 1000 IPM!), meaning the operator must be cautious when verifying programs during rapid motion commands. Fortunately, there is a way for the operator to override the rapid rate during program verification.

The command almost all CNC machines use to command rapid motion is G00. Within the G00 Command, the end point for the motion is given. Control manufacturers vary with regard to what actually happens if more than one axis is included in the rapid motion command. With most controls, the machine will move as fast as possible in all axes commanded.

1.2.2. Straight-line motion (Also called linear interpolation)

This motion type allows the programmer to command perfectly straight-line movements. This motion type also allows the programmer to specify the motion rate (feedrate) to be used during the movement. Straight line motion can be used any time a straight cutting movement

is required, including when drilling, turning a straight diameter, face or taper, and when milling straight surfaces. The method by which feedrate is programmed varies from one machine type to the next. Generally speaking, machining centers only allow the feedrate to be specific in per minute format (inches or millimeters per minute) Turning centers also allow feedrate to be specified in per revolution format (inches or millimeters per revolution).

A G01 word is commonly used to specify straight-line motion. Within the G01, the programmer will include the desired end point in each axis.

1.2.3. Circular motion (also called circular interpolation)

This motion type causes the machine to make movements in the form of a circular path. This motion type is used to generate radii during machining. Two-G codes are used with circular motion. G02 is commonly used to specify clockwise motion while G03 is used to specify counter clockwise motion. Additionally, circular motion requires that, by one means or another, the programmer specifies the radius of the arc to be generated. With newer CNC controls this is handled by a simple "R" word. The R word within the circular command simply tells the control the radius of the arc being commanded. With older controls, directional vectors (specified by J, K, and I) tell the control the location of the arc's center point. Since controls vary with regard to how directional vectors are programmed, and since the R word is becoming more and more popular for radius designation, present examples will show the use of the R word. [Reference: 34]

1.2.4. Example program showing three types of motion.

Program

O0002 (Program number)

N005 G54 G90 S350 M03 (Select coordinate system, absolute mode, and start spindle CW at 350 RPM)

N010 G00 X-.625 Y-.25 (Rapid to point 1)

N015 G43 H01 Z-.25 (Instate tool length compensation, rapid tool down to work surface)

N020 G01 X5.25 F3.5 (Machine ın straight motion to point 2)

N025 G03 X6.25 Y.75 R1.0 (CCW circular motion to point 3)

N030 G01 Y3 25 (Machine in straight motion to point 4)

N035 G03 X5 25 Y4.25 R1.0 (CCW circular motion to point 5)

N040 G01 X.75 (Machine ın straight motion to point 6)

N045 G03 X-.25 Y3.25 R1 0 (CCW circular motion to point 7)

N050 G01 Y.75 ((Machine ın straight motion to point 8)

N055 G03 X.75 Y-.25 R1.0 (CCW circular motion to point 9)

N060 G00 Z.1 (Rapid away from workpiece in Z)

N065 G91 G28 Z0 (Go to the machine's reference point in Z)

N070 M30 (End of program)

1.3. CNC Programming

Almost all current CNC controls use a word address format for programming. By word address format, it is meant that the CNC program is made up of sentence-like commands. Each command is made up of CNC words. Each CNC word has a letter address and a numerical value. The letter address (X, Y, Z, etc.) tells the control the kind of word and the numerical value tells the control the value of the word. Used like words and sentences in the English language, words in a CNC command tell the CNC machine what it is we wish to do at the present time.

The control will first read, interpret and execute the very first command in the program. Only then will it go on to the next command, read, interprete and execute. Then on to the next command. The control will continue to execute the program in sequential order for the balance of the program.

Here is a brief list of some of the word types and their common letter address specifications.

- O - Program number (Used for program identification)
- N - Sequence number (Used for line identification)
- G - Preparatory function
- X - X axis designation
- Y - Y axis designation
- Z - Z axis designation
- R - Radius designation
- F - Feedrate designation
- S - Spindle speed designation
- H - Tool length offset designation
- D - Tool radius offset designation
- T - Tool Designation
- M - Miscellaneous function (See below)

Many of the letter addresses are chosen in a rather logical manner (T for tool, S for spindle, F for feedrate, etc.). There are two letter addresses (G and M) which allow special functions to be designated. The preparatory function (G) specified is commonly used to set modes. Like preparatory functions, miscellaneous functions (M words) allow a variety of special functions. Miscellaneous functions are typically used as programmable switches (like spindle on/off, coolant on/off, and so on). They are also used to allow programming of many other programmable functions of the CNC machine tool.

Certain letter addresses (CNC words) allow the specification of real numbers (numbers that require portions of a whole number). Examples include X axis designator (X), Y axis designator (Y), and radius designator (R). Almost all current model CNC controls allow a decimal point to be used within the specification of each letter address requiring real numbers. For example, X3.0625 can be used to specify a position along the X axis. On the other hand, some letter addresses are used to specify integer numbers. Examples include the spindle speed designator (S), the tool station designator (T), sequence numbers (N),

preparatory functions (G), and miscellaneous functions (M). For these word types, most controls do NOT allow a decimal point to be used

All but the very simplest CNC machines have programmable functions other than just axis motion. With today's full blown CNC equipment, almost everything about the machine is programmable. CNC machining centers, for example, allow the spindle speed and direction, coolant, tool changing, and many other functions of the machine to be programmed. In similar fashion, CNC turning centers allow spindle speed and direction, coolant, turret index, and tailstock to be programmed. And all forms of CNC equipment will have their own set of programmable functions.

The list of programmable functions vary dramatically from one machine to the next, and the user must learn these programmable functions for each CNC machine to be used.

Here is a sample Program:

O0001 (Program number)

N005 G54 G90 S400 M03 (Select coordinate system, absolute mode, and turn spindle on CW at 400 RPM)

N010 G00 X1 Y1 (Rapid to XY location of first hole)

N015 G43 H01 Z.1 M08 (Instate tool length compensation, rapid in Z to clearance position above surface to drill, turn on coolant)

N020 G01 Z-1.25 F35 (Feed into first hole at 35 mm per minute)

N025 G00 Z.1 (Rapid back out of hole) N030 X2. (Rapid to second hole)

N035 G01 Z-1.25 (Feed into second hole)

N040 G00 Z.1 M09 (Rapid out of second hole, turn off coolant)

N045 G91 G28 Z0 (Return to reference position in Z)

N050 M30 (End of program command)

1.4. Turning operation on lathes

1.4.1. Lathes

Lathes are used for the production of all kinds of components which are symmetrical about their axis of rotation. Turning is the machining operation that produces cylindrical parts. In a turning or facing operation on a lathe, the workpiece rotates to provide the cutting motion, and the feed is given by motion of the cutting tool.

In its basic form, it can be defined as the machining of an external surface:

- with the workpiece rotating,
- with a single-point cutting tool, and
- with the cutting tool feeding parallel to the axis of the workpiece and at a distance that will remove the outer surface of the work.

Taper turning is practically the same, except that the cutter path is at an angle to the work axis. Similarly, in contour turning, the distance of the cutter from the work axis is varied to produce the desired shape.

Even though a single-point tool is specified, this does not exclude multiple-tool setups, which are often employed in turning. In such setups, each tool operates independently as a single-point cutter.

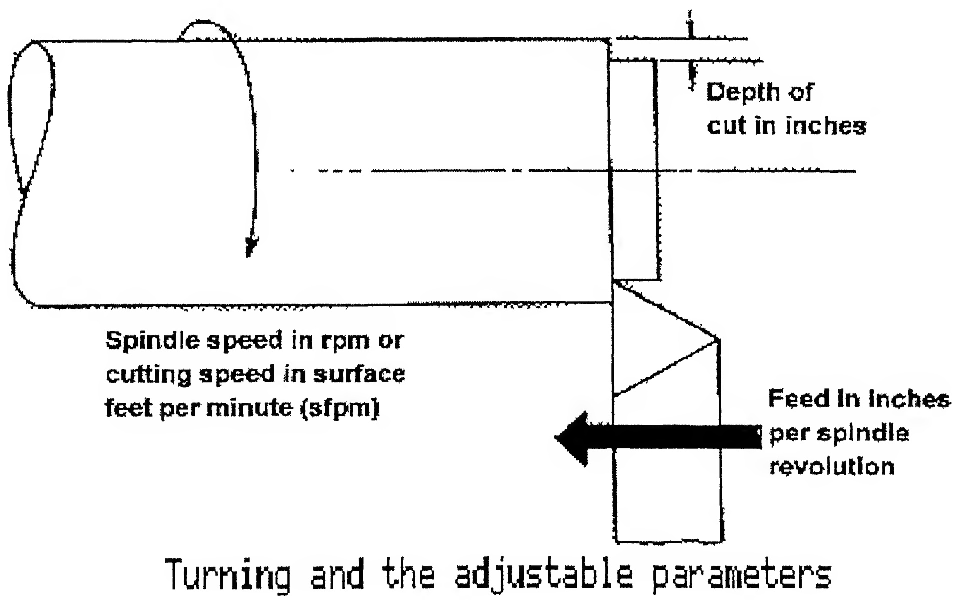


Figure 1.2. Turning ,[Reference 16]

1.4.2. Adjustable cutting factors in turning

The three primary factors in any basic turning operation are speed, feed, and depth of cut. Other factors such as kind of material and type of tool have a large influence, of course, but these three are the ones the operator can change by adjusting the controls, right at the machine.

Speed, always refers to the spindle and the workpiece. When it is stated in revolutions per minute(rpm) it tells their rotating speed. But the important figure for a particular turning operation is the surface speed, or the speed at which the workpeece material is moving past the cutting tool. It is simply the product of the rotating speed times the circumference (in feet or meter) of the workpiece before the cut is started. It is expressed in surface meter per minute (smpm), and it refers only to the workpiece. Every different diameter on a workpiece will have a different cutting speed, even though the rotating speed remains the same.

Feed, always refers to the cutting tool, and it is the rate at which the tool advances along its cutting path. On most power-fed lathes, the feed rate is directly related to the spindle speed and is expressed in inches or mm (of tool advance) per revolution (of the spindle).

Depth of Cut, is practically self explanatory. It is the thickness of the layer being removed from the workpiece or the distance from the uncut surface of the work to the cut surface, expressed in inches or mm. It is important to note, though, that the diameter of the workpiece is reduced by two times the depth of cut because this layer is being removed from both sides of the work.

1.4.3. Lathe related operations

The lathe, of course, is the basic turning machine. Apart from turning, several other operations can also be performed on a lathe.

1.4.3.1. Boring

Boring always involves the enlarging of an existing hole, which may have been made by a drill or may be the result of a core in a casting. An equally important, and concurrent, purpose of boring may be to make the hole concentric with the axis of rotation of the

workpiece and thus correct any eccentricity that may have resulted from the drill's having drifted off the center line. Concentricity is an important attribute of bored holes. When boring is done in a lathe, the work usually is held in a chuck or on a face plate. Holes may be bored straight, tapered, or to irregular contours. Boring is essentially internal turning while feeding the tool parallel to the rotation axis of the workpiece.

1.4.3.2. Facing

Facing is the producing of a flat surface as the result of a tool's being fed across the end of the rotating workpiece. Unless the work is held on a mandrel, if both ends of the work are to be faced, it must be turned end for end after the first end is completed and the facing operation repeated. The cutting speed should be determined from the largest diameter of the surface to be faced. Facing may be done either from the outside inward or from the center outward. In either case, the point of the tool must be set exactly at the height of the center of rotation. Because the cutting force tends to push the tool away from the work, it is usually desirable to clamp the carriage to the lathe bed during each facing cut to prevent it from moving slightly and thus producing a surface that is not flat. In the facing of casting or other materials that have a hard surface, the depth of the first cut should be sufficient to penetrate the hard material to avoid excessive tool wear.

1.4.3.3. Parting off

Parting off is the operation by which one section of a workpiece is severed from the remainder by means of a cutoff tool. Because cutting tools are quite thin and must have considerable overhang, this process is less accurate and more difficult. The tool should be set exactly at the height of the axis of rotation, be kept sharp, have proper clearance angles, and be fed into the workpiece at a proper and uniform feed rate.

1.4.3.4. Threading

Lathe provided the first method for cutting threads by machines. Although most threads are now produced by other methods, lathes still provide the most versatile and fundamentally simple method. Consequently, they often are used for cutting threads on special workpieces where the configuration or nonstandard size does not permit them to be made by less costly methods. There are two basic requirements for thread cutting. An accurately shaped and properly mounted tool is needed because thread cutting is a form-cutting operation. The resulting thread profile is determined by the shape of the tool and its

position relative to the workpiece. The second requirement is that the tool must move longitudinally in a specific relationship to the rotation of the workpiece, because this determines the lead of the thread.

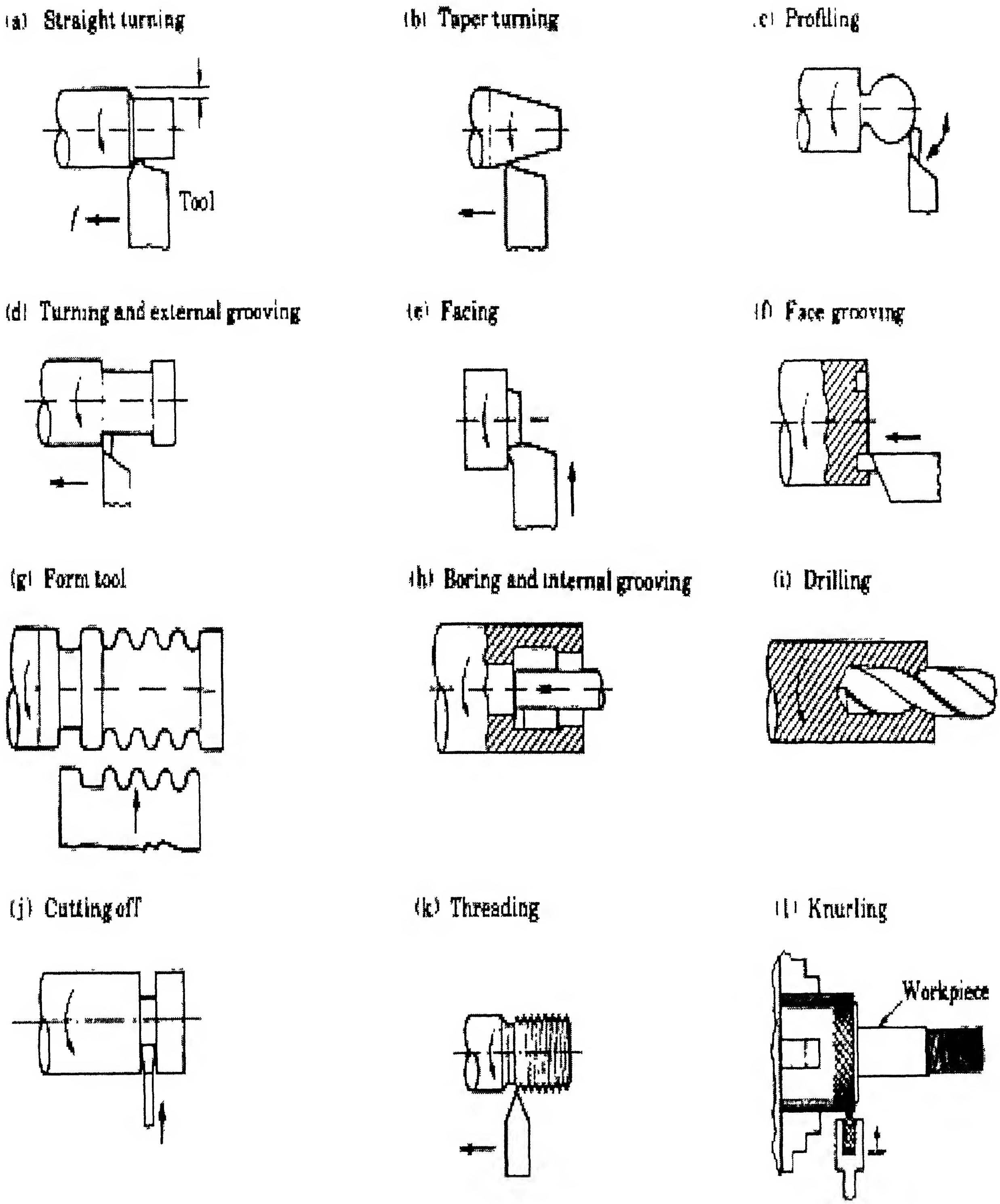


Figure1.3. Operations on lathe, [Reference 30]

1.5. Motivation for the Present work

The ever-increasing use of NC and CNC in industry has created a need for personnel who are knowledgeable about and capable of preparing the programs that guide the machine tools to produce parts to the required shape and accuracy. Dependence on the skilled programmers and the time taken to write and run the programs is significant That's why the computer assisted part programming can be of immense help for the people working in manufacturing industries. Computer aided manufacturing (CAM) software is an important link in the manufacturing process. It forms the connection between computer aided drafting (CAD) designs and CNC manufacturing by incorporating the machining codes within the CAD file. CNC turning machine is perhaps the most widely used conventional machining process. Various CAM systems are available for the part program generation. But most of them are very expensive. Hence there has been a need of a CAM software which is not very difficult to use, yet gives a fairly accurate CNC program. Moreover, since the expertise needed to do the CNC part programming and the time spent is also considerable that's why the need for computer assisted interactive CNC part programming is felt. The present work is an attempt to fulfill these needs.

1.6. Organisation of the thesis

The organisation of the thesis is as follows:

Chapter 2 discusses about some present CAM systems and the related literature.

Chapter 3 discusses in details the design of the proposed computer aided interactive CNC turning part program generation software.

Chapter 4 discusses about the implementation of the proposed application software.

Chapter 5 concludes with the scope of future work.

Chapter 2

Literature survey and the present work

2.1. An overview of CAD

CAD is an acronym for Computer Aided Design. Originally CAD systems were no more than electronic drawing boards. They simply automated the drafting stage of a product's design. These systems were better known as Computer Aided *Drafting* systems. Today CAD has the capability of automating most of, if not the complete, design process. This can lead to compressed design times, which in turn will lead to reduced costs, greater quality, and improved product performance.

CAD systems can be broadly classified into two main categories: these are two-dimensional (2-D) CAD and three-dimensional (3-D) CAD systems.

2.1.1. 2-D CAD

Two-dimensional CAD systems are typically no more than electronic drawing boards. Drawing files created in these systems represent different views of the product in different two-dimensional planes. With these systems it is not possible to check that components fit together in three-dimensional space, it is not possible to determine the properties of the product such as volume, surface area, and centre of gravity, nor is it possible to unfold and fold sheetmetal components.

Often 2-D drawings can be ambiguous and are open to interpretation errors downstream. This is especially true for the more complex designs.

Some of the benefits of implementing a 2-D CAD system are that:

- electronic sketches allow easier editing and copying of features
- it guarantees the quality and reproducibility of drawings
- it should be faster than manual drawing with a proficient operator

2.1.2. 3-D CAD

Three-dimensional CAD is sometimes referred to as geometric modelling. There are three methods of modelling in three dimensions These include *wireframe modelling*, *surface modelling*, and *solid modelling*.

Wireframe modelling represents the part shape with interconnected line elements that provide precise information about edges, corners, and surface discontinuities. Wireframes are the simplest geometric models.

Surface modelling precisely defines the outside part geometry. Surface models overcome some of the ambiguities of wireframes by precisely defining the outside part geometry.

Solid modelling utilises *topology*, that is, the interior volume and mass of an object is defined in the computer in addition to the exterior surface representation. Solid computer models are the most sophisticated forms of three-dimensional modelling.

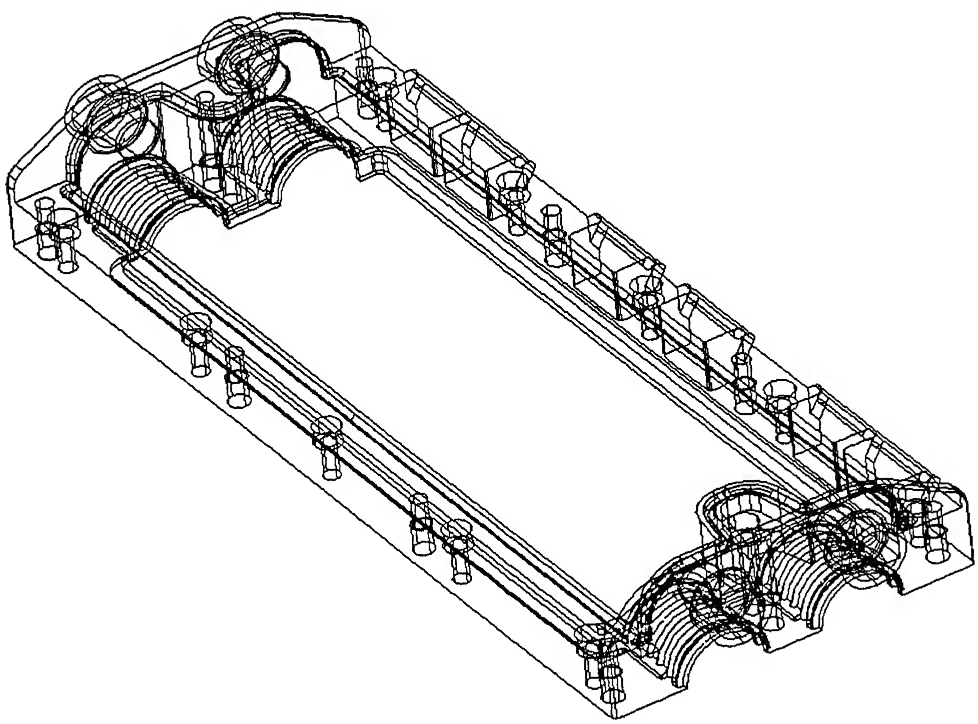


Figure2.1. An Example CAD drawing, [Reference 38]

2.2. A brief history of CAD

The chronological development of CAD is as below:

- The first graphic system was in mid **1950** the US Air Force's SAGE (*Semi Automatic Ground Environment*) air defence system. The system was developed at Massachusetts Institute of Technology's Lincoln Laboratory. The system involved the use of CTR displays to show computer-processed radar data and other information.
- Dr. Patrick J. Hanratty known as " The Father of CADD/CAM" for his pioneering contributions to the field of computer-aided design and manufacturing, developed in **1957** PRONTO, the first commercial numerical-control programming system.
- In **1960** McDonnell Douglas Automation Company (McAuto) founded. It will play a major role on CAD developments with the introduction of CADD program. The first Computer-Aided Design programs used simple algorithms to display patterns of lines at first in two dimensions, and then in 3-D. Early work in this direction had been produced by Prof. Charles Eastman at Carnegie-Mellon University, the Building Description System is a library of several hundred thousands architectural elements, which can be assembled and drawn on screen into a complete design concept. [Reference: 39]
- The conceptual breakthrough of defining objects in terms of 3D reference lines, analogous to the draughtsman's centre line, together with cross-sections normal to them, was produced by S.Matthews, seconded by the Ford Motor Co.
- MCS was founded in **1971** by Dr. Patrick J. Hanratty. Since the day it was founded in **1971**, MCS has enjoyed an enviable reputation for technological leadership in mechanical CADD/CAMsoftware. In addition to selling products under its own name, in its early years MCS also supplied the CADD/CAM software used by such companies as McDonnell Douglas (Unigraphics), Computervision (CADDS), AUTOTROL (AD380), and Control Data (CD-2000) as the core of their own products. In fact, industry analysts have estimated that 70% of all the 3-D mechanical CADD/CAM systems available today trace their roots back to MCS's original code.

- Early solid modeling software first started showing up in the late 70s. Taking basic geometric objects such a sphere, block, cylinders and wedges and combining them using Boolean operations such a remove a cylinder from a block to create a hole. In 1976, MCS introduced AD-2000, a design and manufacturing system for the first 32-bit computers. In **1976** United Computing, developer of the Unigraphics CAD/CAM/CAE system, acquired by Mc Donnell Douglas company. By **1977**, Avions Marcel Dassault assigned its engineering team the goal of creating a three-dimensional, interactive program, the forerunner of CATIA (*Computer-Aided Three-Dimensional Interactive Application*).
- Unigraphics introduced the first solid modeling system in **1981**, UniSolid. It was based on PADL-2, and was sold as a stand-alone product to Unigraphics.
- Autodesk was founded by sixteen people in **April 1982** in California by initiative of John Walker. John Walker has been running Marinchip Systems for two years before. The first version of AutoCAD was based on a CAD program wrote in 1981 by Mike Riddle called **MicroCAD**,the name changed later.
- In **1985** AutoCAD version 2.1 hit the market with 3D capabilities and Polylines command. AutoCAD has won since 1986 "The Best CAD Product" award from PC World magazine every year for the next 10 years.
- In 1991 First version of Visio Technical, an entry-level 2D drawing program from Visio Corporation hit the market.
- The first AutoCAD (Release 12) for Windows platforms. It required 8 MB RAM and 34 MB Hard Drive space for complete installation. The Windows version of AutoCAD includes 36 icons toolbox, allows multiple AutoCAD sessions, separate Render window, support for Windows GUI, DDE and OLE, as well as Drag-and-Drop and Bird's Eye view capabilities. The AutoCAD main menu has been eliminated; After initial configuration, AutoCAD displays the graphics screen. AutoCAD 12 for Windows was one of the most successful CAD programs ever.
- **1995:** AutoCAD View from Autodesk is a CAD tool for viewing and redlining drawings. It can access more than 150 file formats. AutoSurf version 2.1 for use with AutoCAD 13, integrates precise NURBS technology and Advanced Surfacing. Bundled with AutoSurf was the AutoCAD IGES Translator which ensured data

translation across many different CAD systems. AutoCAD Designer release 1.2 from Autodesk, is a parametric solid modeling tool integrated with AutoCAD 12 and 13. It can import AME models and export models created with Designer into 3D Studio. It can calculate mass propriety and moment of inertia. By the end of the year over 350,000 users of Generic CAD worldwide.

- **1997:** SoftSource ships V/Draft, a 2D CAD program launched for Windows with AutoCAD interface, support for **Visual Basic**, C++ and ActiveX programming. AutoCAD Release 14 for Windows was proved to be the best and more stable version ever. It comes with a lot of improvements in both 2D and 3D. Includes ObjectARX 2.0, ActiveX Automation Interface which allows **Visual Basic** for Application to drive AutoCAD or to write a single application that integrates AutoCAD with other ActiveX Automation compliant application, AutoVision is included. TurboCAD Professional version 4 from IMSI.
- **2000:** CoCreate Software Inc., a subsidary of Hewlett Packard Company, announces release of Solid Designer 2000 with five new modules. DataCAD 9, 2D and 3D architectural package, the new version offers support for AutoCAD 2000 DXF and DWG files, can export in WMF and Adobe Acrobat PDF formats. [Reference: 36]

2.3. AutoCAD2000: An introduction

Out of the many CAD softwares for the drafting purpose AutoCAD is the **most versatile** and the most widely used software. AutoCAD® 2000 is a product with a rich and unique history. First released in **1982** under the name MicroCAD, the first AutoCAD ran under the CP/M operating system on Intel 8080 computers. That first AutoCAD release started a revolution in drafting and design. Today, AutoCAD is translated into 18 languages and used by millions of users worldwide on computers a thousand times more powerful than those early 8080 microprocessors.

AutoCAD2000 continues having the AutoCAD heritage. The AutoCAD 2000 interface is more design-centric and less command-centric, making the software more transparent in the design process. In Microsoft Visual Basic® for Applications (VBA),

one can now refer a project from another project. One can also create libraries of common functions and macros.

2.3.1. DXF files in AutoCAD2000

The DXF format is a tagged data representation of all the information contained in an AutoCAD® drawing file. Tagged data means that each data element in the file is preceded by an integer number that is called a group code. A group code's value indicates what type of data element follows. This value also indicates the meaning of a data element for a given object (or record) type. Virtually all user-specified information in a drawing file can be represented in DXF format.

This is for drawing interchange format. DXF is a text file containing drawing information that can be read by other CAD systems or programs. If any one works with consultants who use a CAD program that accepts DXF files then one can share a drawing by saving it in DXF format. One can save a drawing in Release 12, Release 13, Release 14, or AutoCAD 2000 in DXF format. An ASCII or binary file format of an AutoCAD drawing file for exporting AutoCAD drawings to other applications or for importing drawings from other applications. One can also specify a precision of up to 16 places for floating-point numbers and save the drawing in either ASCII or binary format. If a designer do not want to save the entire drawing, he can save selected objects. This feature helps remove extraneous material when the designer want to save only specific blocks or areas of the drawing.

In the DXF format, the definition of objects differs from entities: objects have no graphical representation and entities do. For example, dictionaries are objects not entities. Entities are also referred to as graphical objects while objects are referred to as nongraphical objects. Entities appear in both the BLOCK and ENTITIES sections of the DXF file. The use of group codes in the two sections is identical. Some group codes that define an entity always appear, others are optional and appear only if their values differ from the defaults. The end of an entity is indicated by the next 0 group, which begins the next entity or indicates the end of the section.

2.3.2. Types of DXF files

DXF files can be either ***ASCII*** or ***binary*** formats. Because ASCII DXF files are more common than the binary format, the term DXF file is used to refer to ASCII DXF files and the term binary DXF file for the binary format.

2.4. An overview of CAM

CAM is an acronym for Computer Aided Manufacturing. A CAM system uses the data generated by the CAD system to generate NC code that is required to operate a CNC machine. CAM software is intended to simplify the task of programming machine tools. It enables users to graphically define part geometry and establish machining strategies, develop and verify tool paths, postprocess the cutter location (CL) files, and exchange this NC data with other shop-floor machines.

When considering CAM packages it is first necessary to understand the needs and requirements. Needs should be evaluated using a number of criteria including:

- The level of integration required with the design model
- The need for integrated tool design
- The ability to import foreign data
- Customisation requirements.

Since there is a close relationship between the CAD model and the CAM software generated NC code. The integration of CAD and CAM system is the most desirable. This system is logically named a CAD/CAM system. There are many clear advantages to have a fully integrated system, these include:

- Common user interface which makes learning the software much easier for the CAD operators
- No difficulties in transferring the CAD data to the CAM system

Some disadvantages of integrated CAD/CAM systems are:

- They are typically more expensive than stand alone or add-on packages

- They are sometimes not as flexible or have fewer features than stand alone packages. This is because often the manufacturers of integrated CAD/CAM packages focus more energy on the CAD aspect of their particular machine.

2.4.1. A brief history of the CAM systems

The history of computer aided systems is mainly the history of computer design and constructing thus it is at the beginning joined with history of computer graphics. It was not possible to create, to research and to manipulate with solids by computer before the computer graphics was introduced. Computer aided design and constructing was reduced only to computing.

First design of calculating machine - calculator with graphics options was introduced by Vannevar Bush in 1945, but his proposal has not ever been realized in practice. First computers which allowed interactive graphics were made at the beginning of the 60th years thanks to General Motors, Lockheed, NASA and Bell Labs. First systems of computer graphics and computer aided design and constructing were developed for various hardware platforms, but the biggest producers of computer equipment (IBM, DEC, Control Data, Texas Instruments) have ignored this area. First proposals of these systems were often realized by customers. A lot of work was performed by famous research centres, for example Masachusets Institute of Technology, University of Utah and Xerox Parc in California. There were developed several tenth of systems which solved the tasks of computer aided design in various areas and which were different by quality and completeness. Gradually, about fifteen of the solutions have also been seriously used outside of the developing workplaces. The beginning of the 80th years was carried by the Unix suceeding the older proprietary systems and CA systems which did not adapt to this trend either disappeared, or rapidly yielted from their positions. 80th years are characterized by dominating of big companies which produce sophisticated software systems for computer aided design and later also for following technical processes. Cooperation between customer and company providing CA systems was very close, some of the companies have supported only one customer, respectively were created as department of big industrial giants. Europe did not want to lag behind the development in the U.S. and mainly in France appeared companies interested in

developing the computer aided systems and some of them have important position in the area of CA systems at present.

History of computer aided manufacturing becomes from 50th years, when the conception of numerically controlled machines was designed. This was first impulse for entrance of electronics and later computer technology into the production support. However, more extensive development of computer aided manufacturing systems was enabled by creating the conception of the computer numerically controlled production machines, which is dated to 1970. Since the CAM systems allow the data about product geometry, which has been created by CAD system, to be used directly for creating NC programmes for NC and CNC production machines; the big systems covering the areas of computer aided design and also following computer aided manufacturing appeared at 80th years; these systems are so-called CAD/CAM.

The most important company, which produced extensive and expensive CAD/CAM system was Computervision, that practically dominated in the areas of aircraft and automobile industry. IBM has developed own CAD/CAM system, which was later united with CATIA system. At the beginning of 90th years six companies are strongly entering to foreground by producing the systems working under Unix close by prices and quality: four from U.S. (Computervision, EDS/Unigraphics, SDRC, PTC) and two from France (Matra Datavision and Dassault Systemes, where the majority is owned by IBM). These companies have created modular systems determined mostly for area of the machinery with price several hundred thousands U.S.$ for full (hardware and software) working place. These companies are also dominating in the area of big CAD/CAM systems at present. [Reference: 36]

Area of personal computers (Personal computer - PC) was not attractive for CAM systems for long time. It was mainly due to the fact, that CAD systems for PC were designed only for 2D drawing in regard to their small calculating performance and these did not dispose with functions for creating of solid models, from shape of which is possible to derive programmes for NC production machines and also it did not allow to create the NC programmes effectively, because of their small calculating perfomance. At the beginning of 90th years the big CAD/CAM systems working on workstations were only available for the area of computer aided manufacturing. Calculating and graphical

performance of the machines working under Unix has been increased much rapidly against the PC, mainly thanks to products of Silicon Graphics during 1990 - 1994. Until uncommon development in the area of developing the PC components during the second half of 90th years, with entering the PC with processors Pentium, Pentium Pro, Pentium II, which are competitive by performance to workstations, but their price is only partial, has enabled to the developers of CAM systems to be oriented also to the area of PC and thus enable benefits of CAM to more users.

2.5. Integration of CAD/CAM

Computer Integrated Manufacturing (CIM) means complete integration of all aspects of manufacturing utilizing computerized information. CIM system means the use of component data created with CAD in the CAM environment. In the other words, the part geometry for manufacturing use in computerized form is used for NC programming. This stage of development may be termed small-scale integration. The most highly developed form of CIM is the creation of a database containing all the information required for flexible manufacturing of components produced by the plant, in a form in which it can be retrieved and used by anyone who needs it. Flexible manufacturing means the ability to make any components in small numbers as well as large, quickly, at economical cost, thus reducing tool charge ones, work in process and costly inventory. CIM also provides for inclusion of quality systems and controls in the manufacturing process, rather than applying them afterward.

A major problem in integrating Computer Aided Design (CAD) and Computer Aided Manufacturing (CAM) arises from the difference in thinking between the design and manufacturing people. Designers think of designing a new product in terms of its intended function whereas manufacturing engineers think in terms of decomposing a product design into a set of manufacturing operations. Computer aided manufacturing (CAM) software is an important link in the manufacturing process. It forms the connection between computer aided drafting (CAD) designs and CNC manufacturing by incorporating the machining codes within the CAD file. CAD/CAM systems could produce computerized instructions for computerized machine controllers: lathes, mills, machining centers, turret punches, welding equipment, automated assemblies, etc. A

typical design involves producing part drawings in a CAD program right up to completion of design and making layers of the geometry required for the CAM processing software. The description of part created in a CAD program is translated into an appropriate format, such as **DXF** or IGES, and then loaded into the CAM program which are used then to create tool paths that trace this description. This path can be edited and combined with other tool path files where necessary and the combined forms a complete program for the machine tool to manufacture the part. The resulting NC program can be exported back into the CAD system to produce a simulated backplot of the toolpath or imported into a solid modeling NC program to produce a computer model for checking before manufacturing.

2.6. Development of CNC machines

Numerical control can be defined as an operation of machine tools by means of specifically coded instructions to the machine control system. Numerical control technology as it is today, emerged in the mid 20^{th} century. It can be traced to the year of 1952, the U.S. Air Force, and the names of John Parsons and the Massachusetts Institute of Technology in Cambridge, MA, USA. The real boom came in the form of CNC, around the year of 1972, and a decade later with the introduction of affordable microcomputers. In the manufacturing field, and particularly in the area of metal working, Numerical Control technology has caused something of a revolution.

The first benefit offered by all forms of CNC machine tools is **improved automation**. The operator intervention related to producing workpieces could be reduced or eliminated. Many CNC machines can run unattended during their entire machining cycle, freeing the operator to do other tasks. This gives the CNC user several side benefits including reduced operator fatigue, fewer mistakes caused by human error, and consistent and predictable machining time for each workpiece. Since the machine will be running under program control, the skill level required of the CNC operator (related to basic machining practice) is also reduced as compared to a machinist producing workpieces with conventional machine tools.

The second major benefit of CNC technology is **consistent and accurate** workpieces. Today's CNC machines boast almost unbelievable accuracy and repeatability

specifications. This means that once a program is verified, two, ten, or one thousand identical workpieces can be easily produced with precision and consistency.

A third benefit offered by most forms of CNC machine tools is **flexibility**. Since these machines are run from programs, running a different workpiece is almost as easy as loading a different program. Once a program has been verified and executed for one production run, it can be easily recalled the next time the workpiece is to be run. This leads to yet another benefit, fast change-overs. Since these machines are very easy to setup and run, and since programs can be easily loaded, they allow very short setup time. This is imperative with today's Just-In-Time product requirements.

2.7. GUI (The way for interactive programming)

2.7.1. An Introduction

An interactive programming is an interface for issuing commands to a computer utilizing a pointing device, such as a mouse, that manipulates and activates graphical images on a monitor.

2.7.2. Graphical User Interface (GUI)

GUI is a program that responds to user activity where two way information flow takes place and is acting or capable of acting on each other. It is a term describing a program whose input and output are interleaved, like a conversation, allowing the user's input to depend on earlier output from the same run.

DOS, UNIX and other command-line operating systems have long been criticized for the complexity of their user interface. This interface evolved in the late 60's and early 70's with the emergence of timesharing as the new leading computing style. Using terminals on shared minicomputers, UNIX evolved a set of applications that shared a common on-oriented interface. This interface was revolutionary in its time for its simplicity and power. UNIX introduced the concept of the shell, a command interpreter that read common lines from the keyboard and executed separate process for each command, emerged in the late 70's. Researchers at XeroxPARC were doing things differently. They were experimenting with GUIs that replaced the character display and command line with a large bit-mapped display; icons, multiple windows, and a pointing

device call a mouse, developed earlier at SRI. Research showed that people could learn to use applications with a graphical interface more quickly than with commands. The graphical interfaces were also easier to remember and helped users get more done quickly and they where more user friendly. In a GUI environment, input options to many computer programs are designed as a set of icons, which are graphic symbols that look like processing option they are meant to represent. Users of such programs select processing options by using a mouse or similar device to point to the appropriate icon. In the early days when fools argued that a command-line structure was faster or more efficient than a GUI, They called GUI users WIMPs - Window, Icons, Mice and Pointer users.

The developers of GUIs also went to great lengths to provide the user with immediate, visual feedback about the effect of each action. A simple consistent command language with abundant memory aids, and immediate feedback is what makes a GUI easier to use. The application's interface should operate smoothly and predictable, anticipating what the user will do next. In the same context, a poorly designed interface will be noticed and will be unappreciated by the user.

Some CAD/CAM softwares are CadMax, Anvil 5000, Microstation Bentley, Unigraphics, Esprit, Pro/Engineer, Solid-Edge, SolidDesigner, Catia, ParaSolid, MasterCam, AutoCad(Autodesk), Power station, surfcam, ArchiCAD, Autocode, intelliCAD etc.

For having a feel of the interface the start up screen of one of the software (Power station) is shown here.

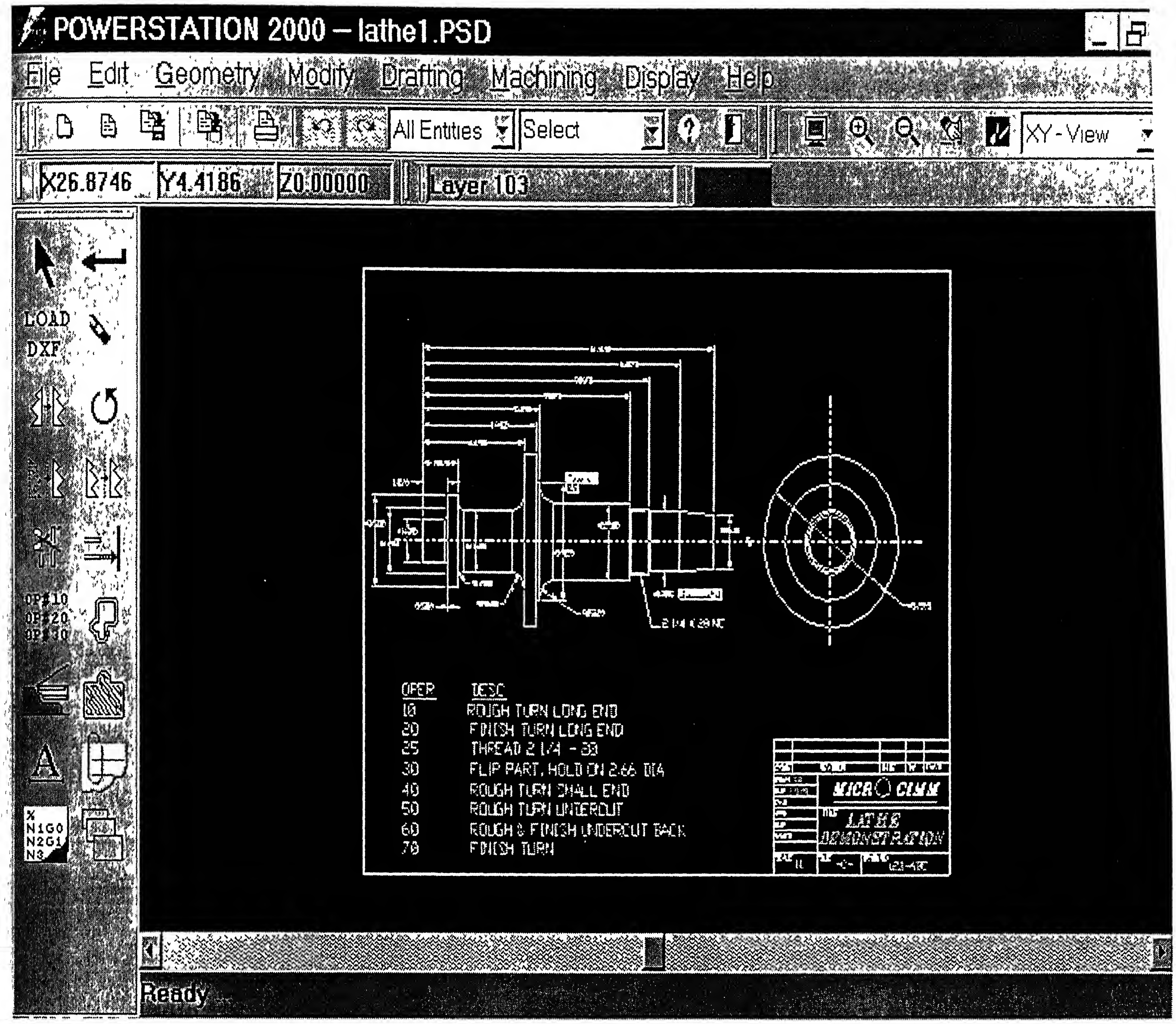


Figure 2.2. power station start up screen,[Reference 35]

It is a completely integrated 2 1/2 axis CAD/CAM Programming system for Mills, Lathes, EDM's and Waterjet/Flamecutters. Utilizes an easy-to-use mouse/menu interface that includes DXF file import/exporting.

2.8. Present work

The software has been developed in Visual Basic, under windows operating system. The system uses the two dimensional Geometry drawn in AutoCAD with Feature oriented Approach, which is suited to simultaneous design and process sequencing. Standard features are recognised by the software and the corresponding coordinates are also read automatically. Lastly the CNC part program is generated, where coordinates of the features correspond to stock removal material volumes. The tool path animation is also incorporated to help the user verify the correctness of the CNC part program.

The software firsly asks to input the drawing name. After the drawing name is inputted the software guides the user for the rest of the session. The software runs interactively and the input data is keyed into the system during a user-friendly interactive session.There is option for the user to input the coordinates of the standard features manually or recognise them automatically just by clicking the mouse. The work piece and tool material is then chosen from the list of materials. The size of the blank is then inputted geometrically by its length and diameter which are used to generate the cylindrical primitive. All the time, user interacts with the software by keying appropriate values of parameters to describe the machined features of the component. Description of the workpiece material and features are written to output files and to outline process sequencing. For each feature its coordinates and other essential informations are saved in automatically made files. The user can see these informations at any stage while running the application software.

Chapter 3
Design of the proposed interactive CNC part program generation software
3.1. Introduction

The salient features of the application software are given in this section. In the present work, it is proposed to design and develop an interactive part program generating software which also animates the tool path of the CNC turning machine. It is proposed to have the following modules in the system:

- Input of the part geometry in AutoCAD
- Checking the standard features in the part to be manufactured
- Input of the blank size dimensions and checking whether the size of the blank is large enough so that the required job can be obtained.
- Input of other parameters, such as work piece material, tool material and dimensions of the special features by viewing AutoCAD drawing.
- Getting the CNC part program and viewing animation of the tool path in the form of graphics.

3.1.1. Input of the part geometry

The user is required to make the drawing of the part. Here the point is to be noted that the user should draw only the standard features of the drawing for rough turning operation. These standard features include

(a) face up

(b) face down

(c) taper up

(d) taper down

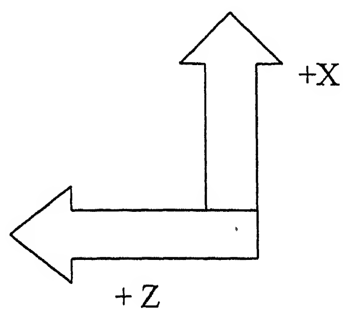
(e) cylindrical surface

(f) groove

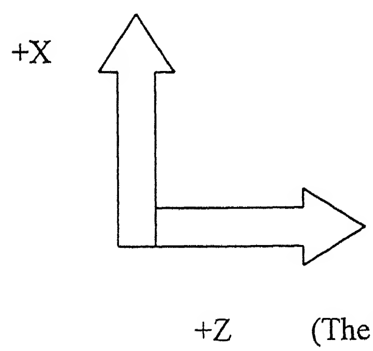
A Cartesian coordinate system has three axes: X, Y, and Z. When coordinate values are, entered it indicates a point's distance (in units) and its direction (+ or –) along the X, Y, and

Z axes relative to the coordinate system origin (0, 0, 0) or relative to the previous point Usually, when a new drawing is started in AutoCAD®, users are automatically in the world coordinate system (WCS); the X axis is horizontal, the Y axis is vertical, and the Z axis is perpendicular to the XY plane.

In addition to the WCS, one can define a movable user coordinate system (UCS) with a different origin and axes in different directions.The user can define a UCS in terms of the WCS . For the convinience of the user, the world coordinate system (WCS) of AutoCAD should be changed into the user coordinate system. Then the direction of the X and Y axes should be changed in the way shown below. The drawing should be made keeping in mind the initial rough turning operations on CNC lathe.



(User coordinate system)



(The World cordinate system in AutoCAD)

Figure 3.1. WCS and UCS

Here is an example illustrating a drawing:

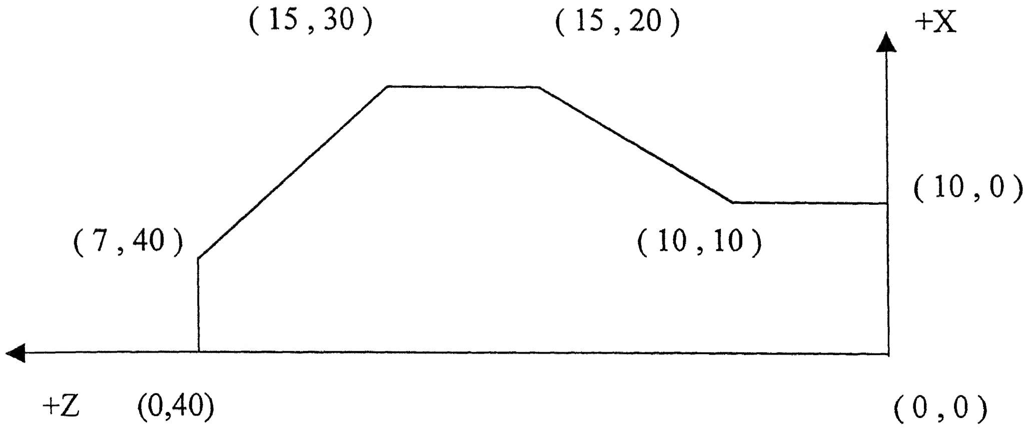


Figure 3.2. Drawing

After making the drawing the user is required to save it in .dwg and .DXF file extension.

3.2. Reading the drawing

Now the software is ready for reading the drawing and doing furthur operations. Just after mouse clicking run command the start up screen(form) appears. This displays the title and vital specifications about the software. Help are included I the form of tool tip text. Whenever the user places the cursor on any label the information about that label is displayed for quick reference.

When the user clicks the start command the main screen having various menus appears.

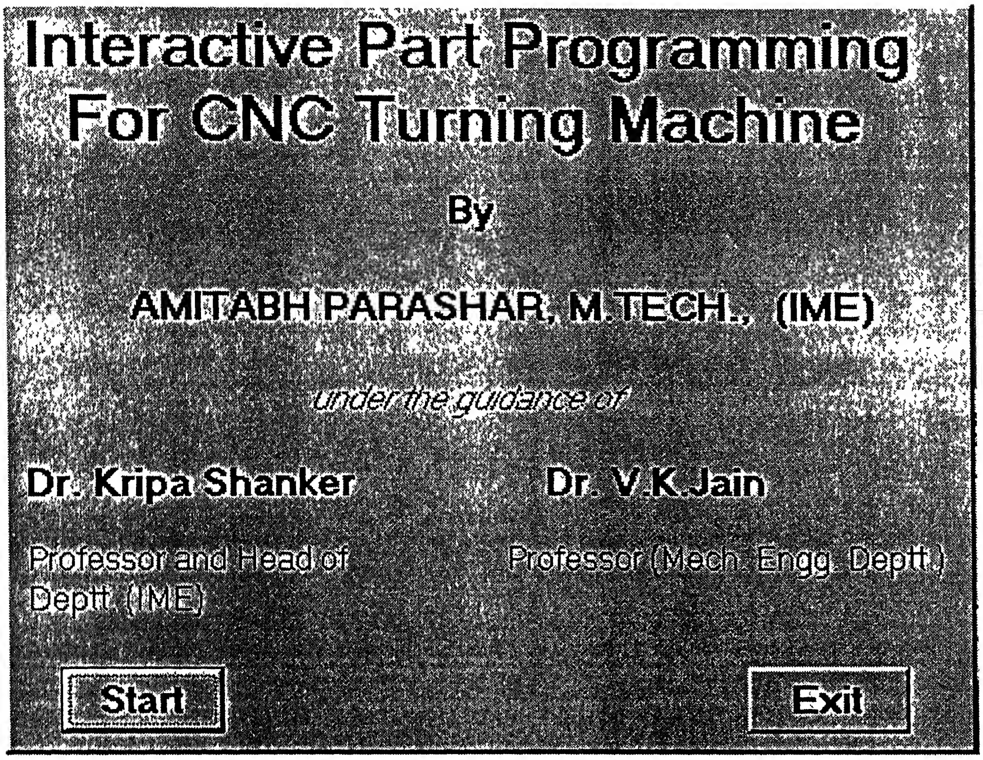


Figure 3.3. Start up screen

The first menu is ***Demo***. This is an introductory animation to give the new users the feel of how a CNC turning machine cuts. The user is required to choose the feed and depth of cut and accordingly the material is cut by the tool. The animation gives response for the change of feed or depth of cut which is quite obvious by viewing the animation.

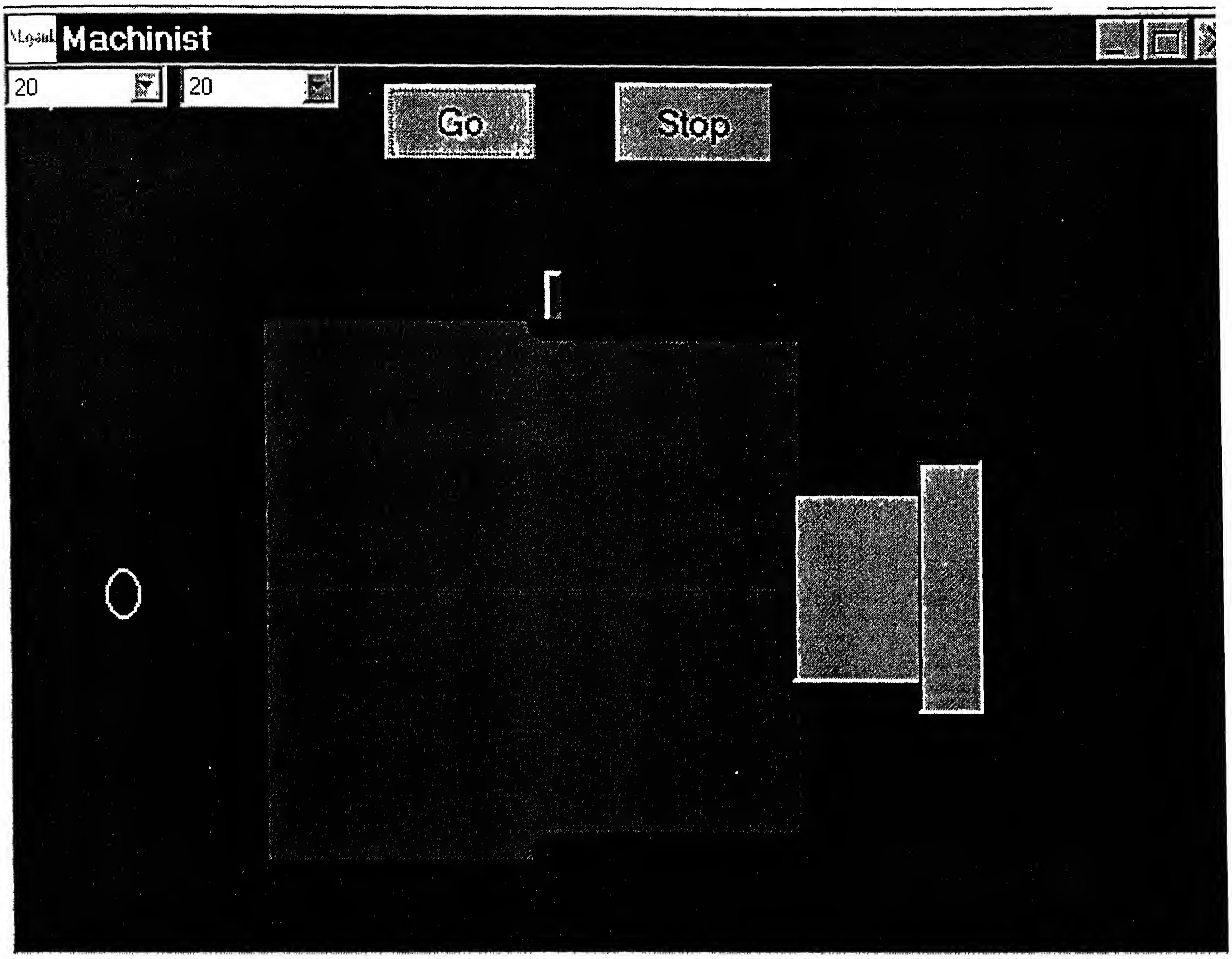
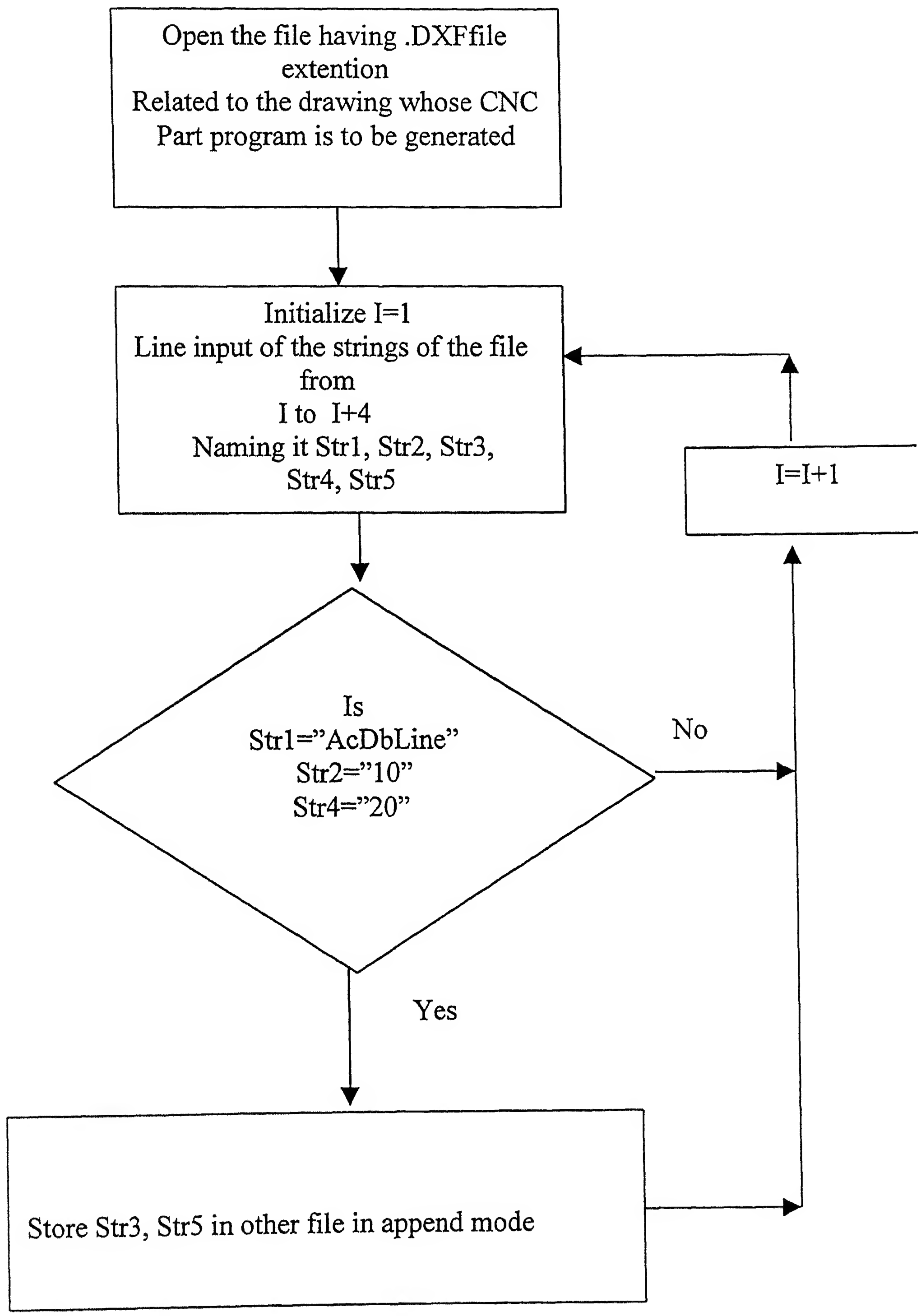


Figure 3.4. Turning operation animation

The second menu is ***File***. When the user clicks this File menu a dialog box appears from which the AutoCAD drawing can be browsed. The drawing whose CNC part program is to be obtained is selected in .DXF file extension. The drawing is read by the software and is displayed on the main screen.

Algorithm for reading .DXF File from AutoCAD byVisual Basic software



3.3. Recognising the standard features of the drawing

The next menu is ***Features Statistics***. This tells about the standard features in the drawing. Total number of various features are displayed on the screen. The drawing is also displayed on the screen. The user can verify by visualising the drawing that all the standard features of the drawing is read by the software.

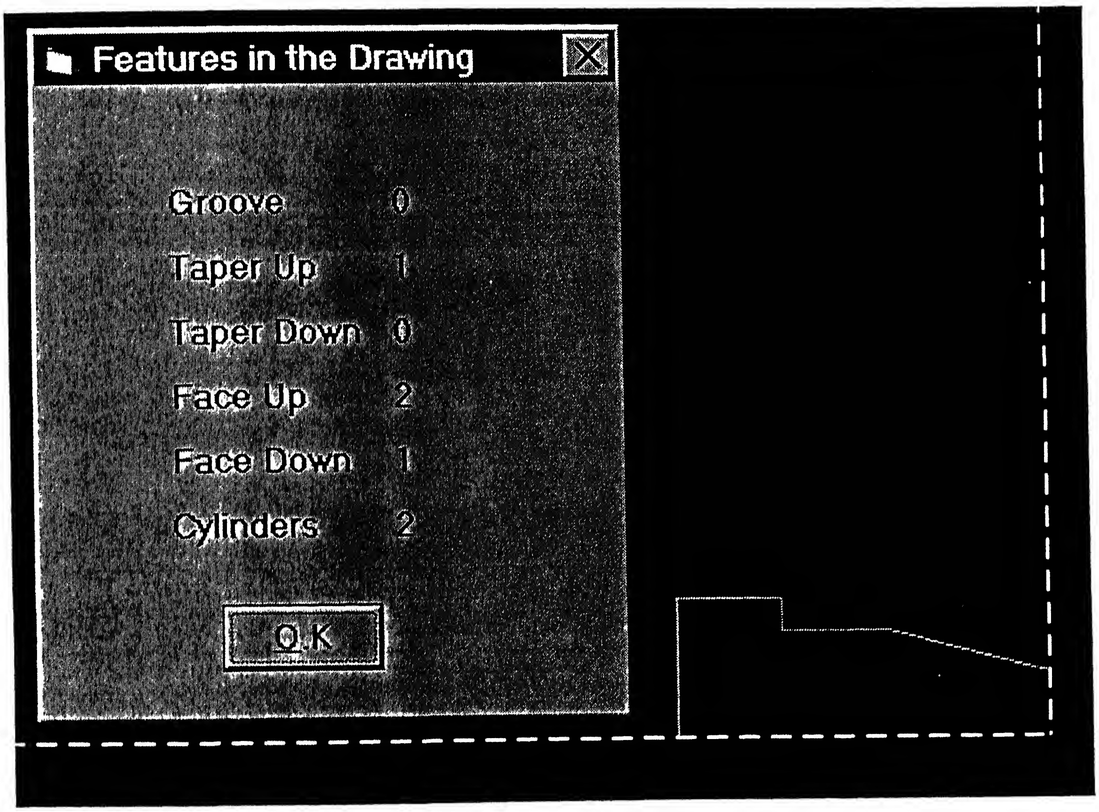
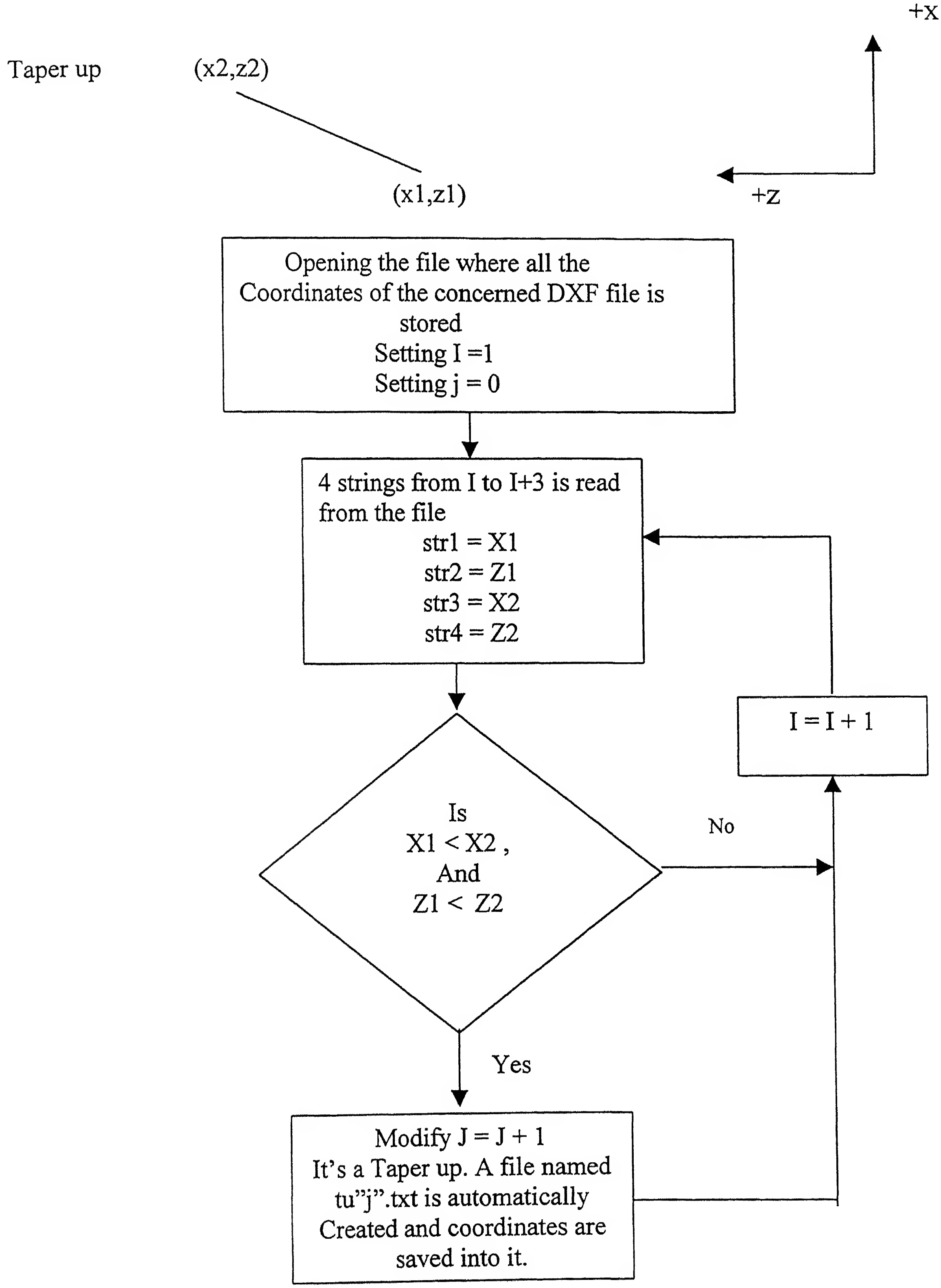
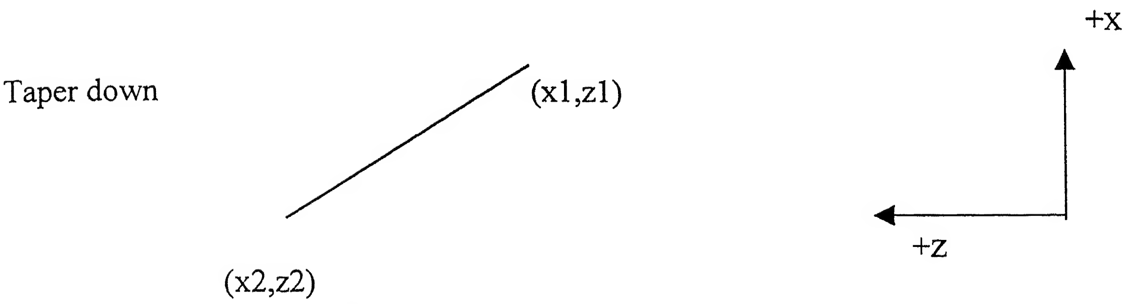


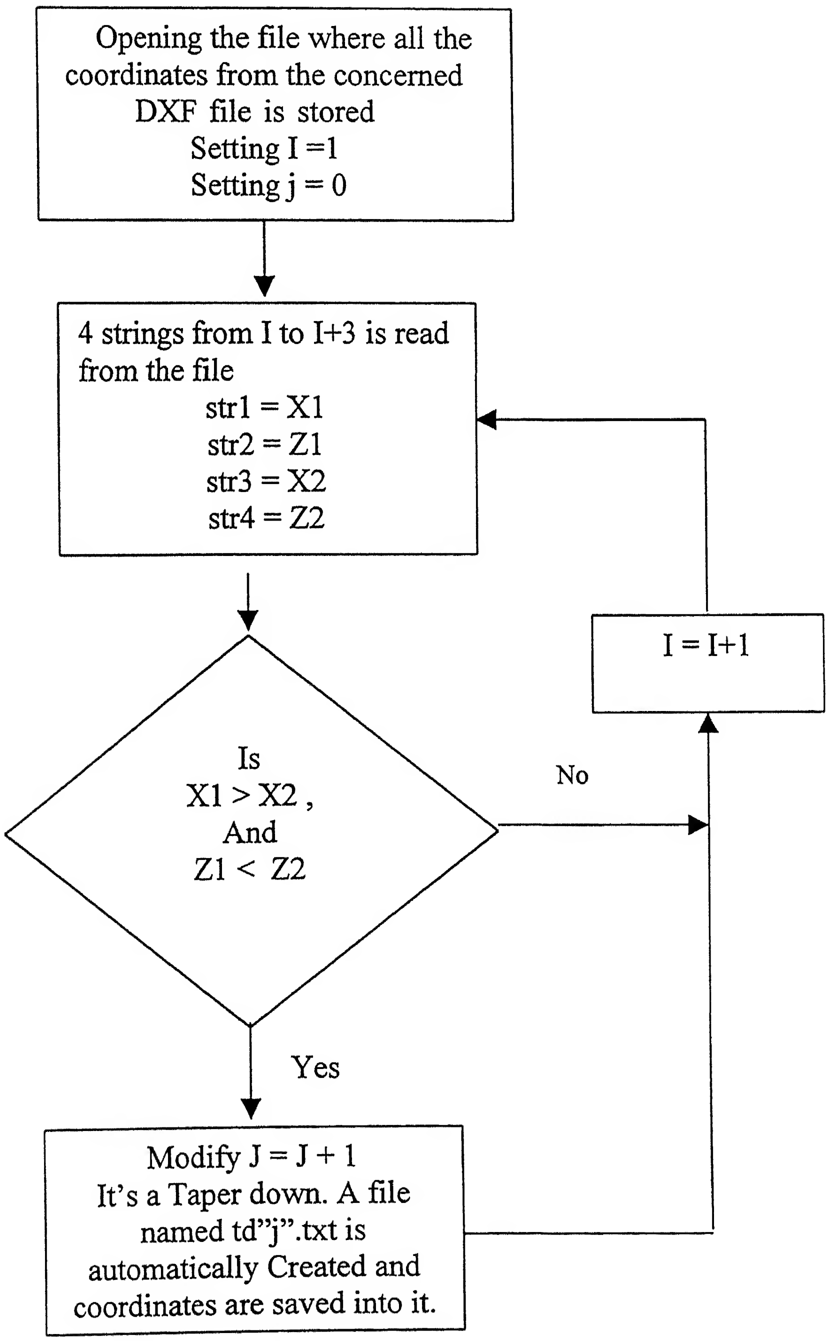
Figure 3.5. Reading the features of the drawing

Feature Recognition Algorithm (Taper up)

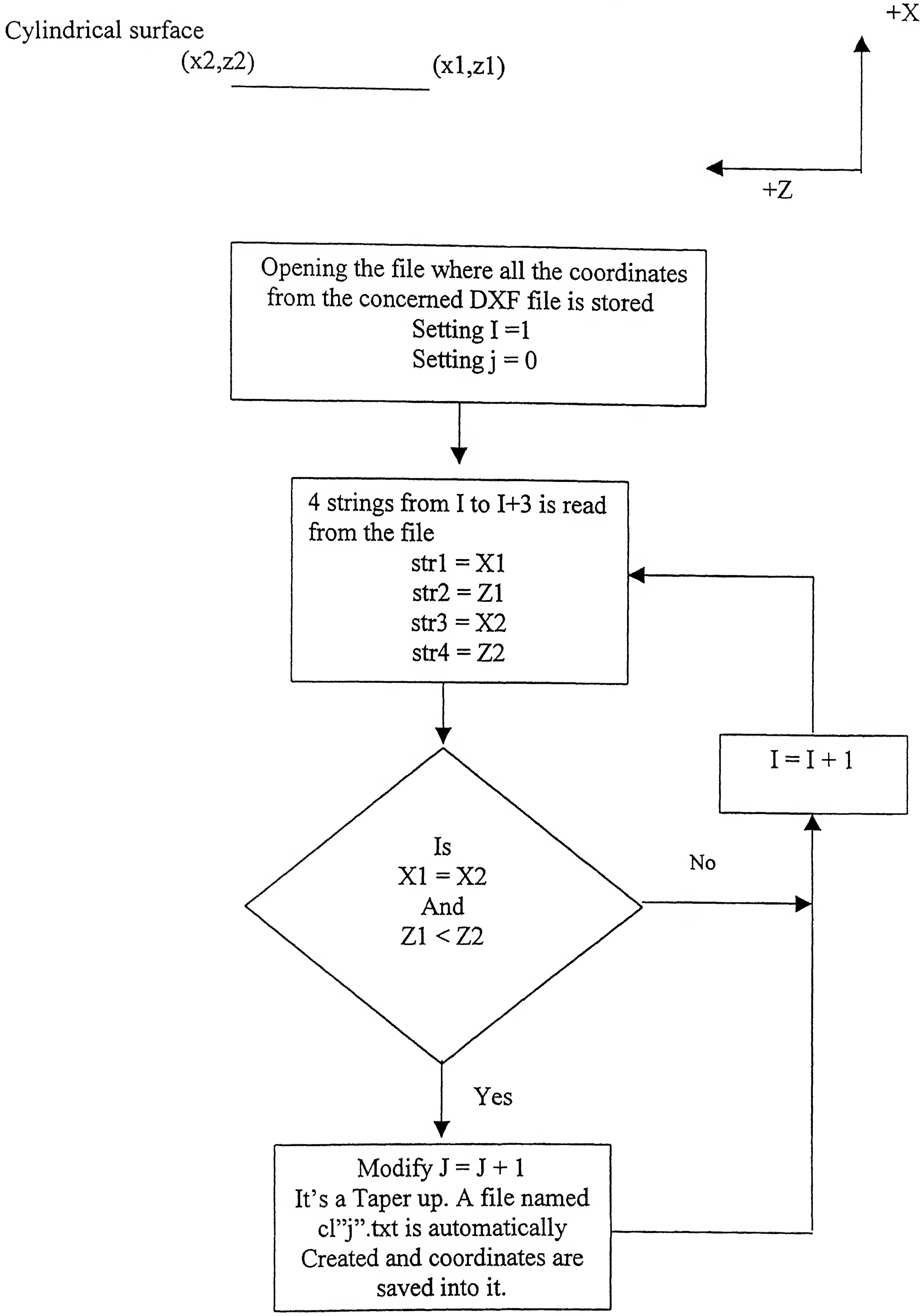


Feature Recognition Algorithm(Taper down)



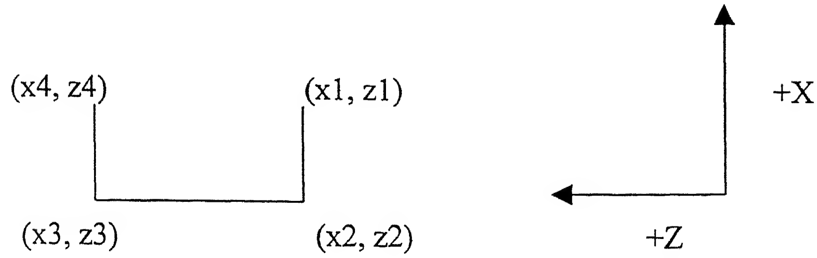


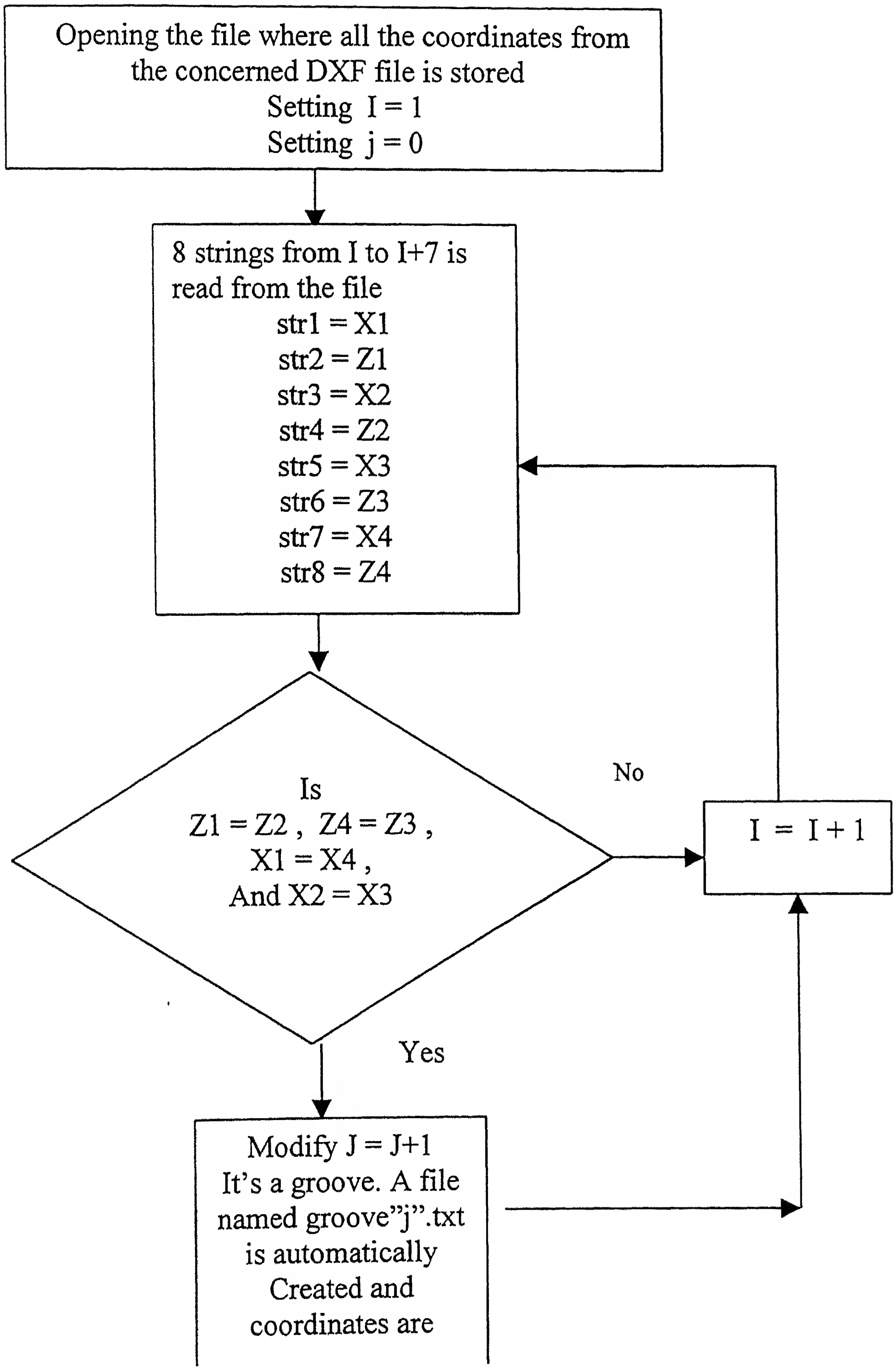
Feature RecognitionAlgorithm (cylinder)



Feature Recognition Algorithm (groove)

For groove:





3.4. Feeding the dimensions of the blank and choosing material

The next menu is named as ***Process***. When it is clicked another screen appears which prompts the user to input the blank size. When the user clicks this command another screen appears in which the diameter and the length of the blank is inputted. After that the tool material and the work piece material is chosen just by the mouse clicking. The program compares the dimensions of the actual workpiece and the blank size. If there is any anomaly a message box appears which tells the user to change the workpiece. If there is no anomaly the process sequencing takes place and the corresponding CNC part program for rough material removal is generated. To get the rough idea about the amount of material removed the next menu ***Animate*** can be clicked. Speed, feed and depth of cut are auomatically selected depending upon the workpiece.

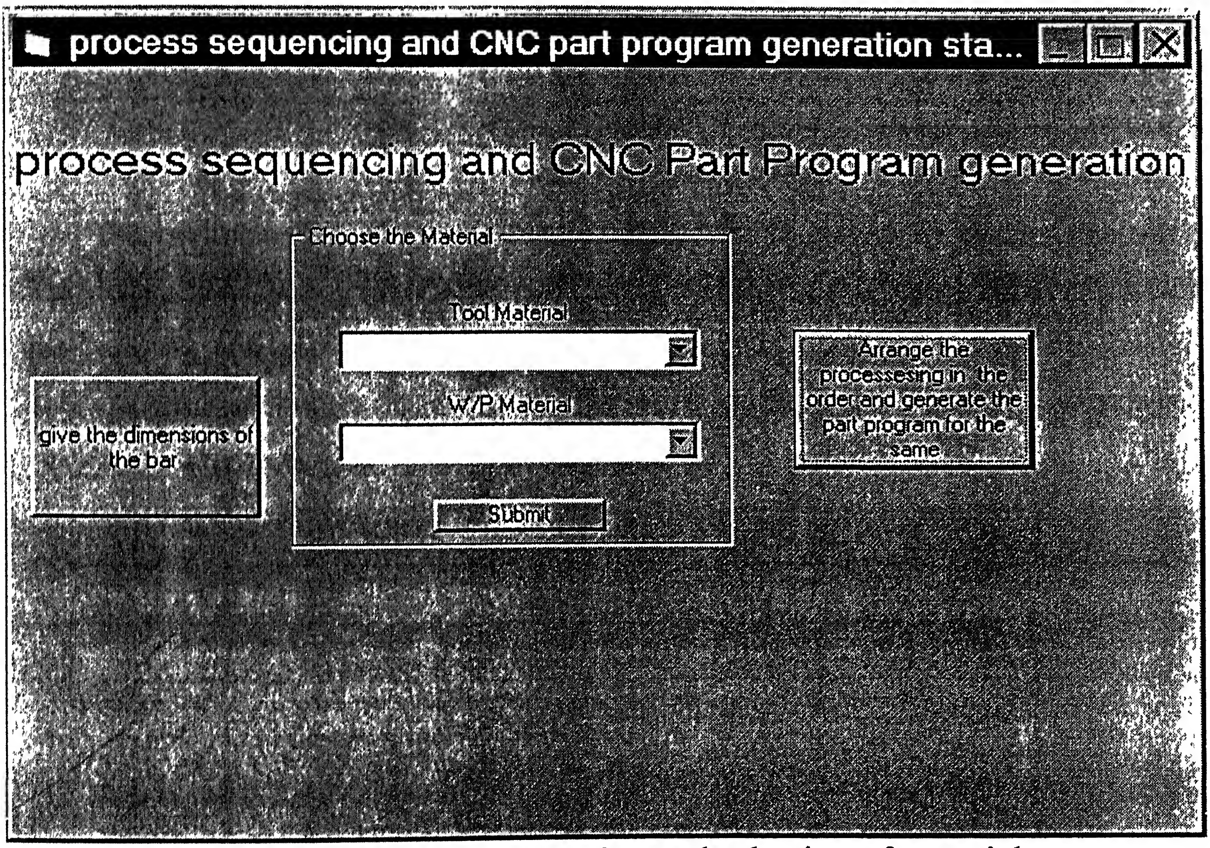


Figure 3.6. Process sequencing and selection of materials

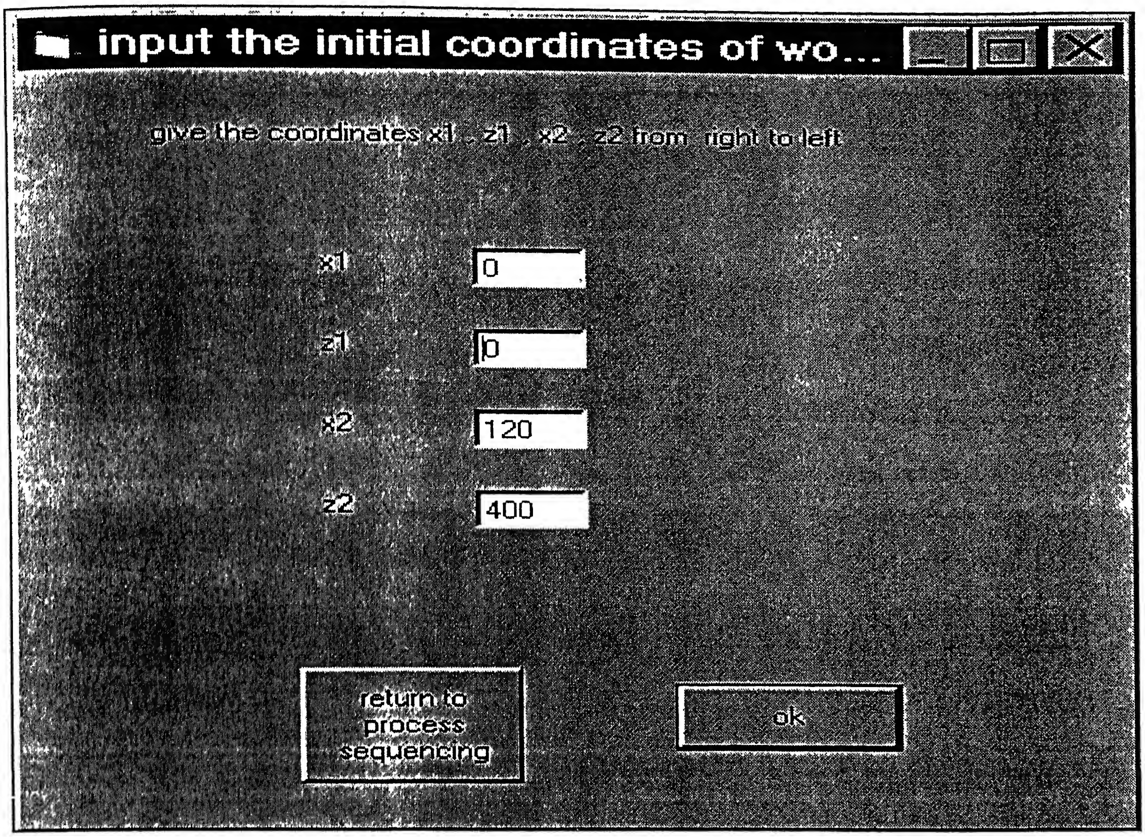


Figure 3.7. Dimensions of tha blank

The next step is to define the real contour of the workpiece by clicking the menu named ***Contour(Turning Cycle)***. When this menu is clicked a new screen appears in which there are two options line and arc. The user is required to define the complete contour. Dimensioned sketch of the line and the arc are placed on the screen to assist the user. When the contour is defined fully, OK command is clicked.

The next menu is meant for inputting the coordinates for special features and is named ***Additional Features.*** When this menu is clicked a screen appears and prompts the user to input the dimensions of the special features. To assist and direct the user this screen is also loaded with pictures. After inputting all the concerned data the user is required to click finish command at last.

Next menu is named ***Turning***. Tool path animation can be seen which is based on the CNC part program which has just been generated. The part program is stored and can be retrieved from a separate file. The user can verify the tool path by viewing it.

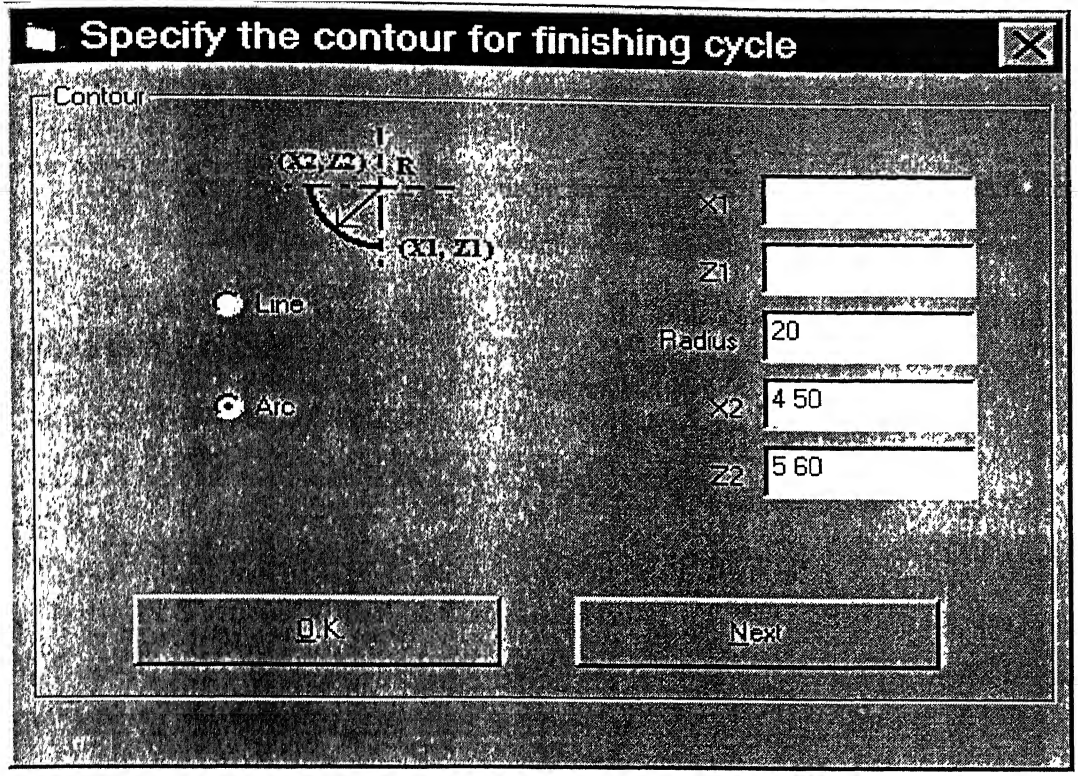


Figure 3.8. Specify the Arc

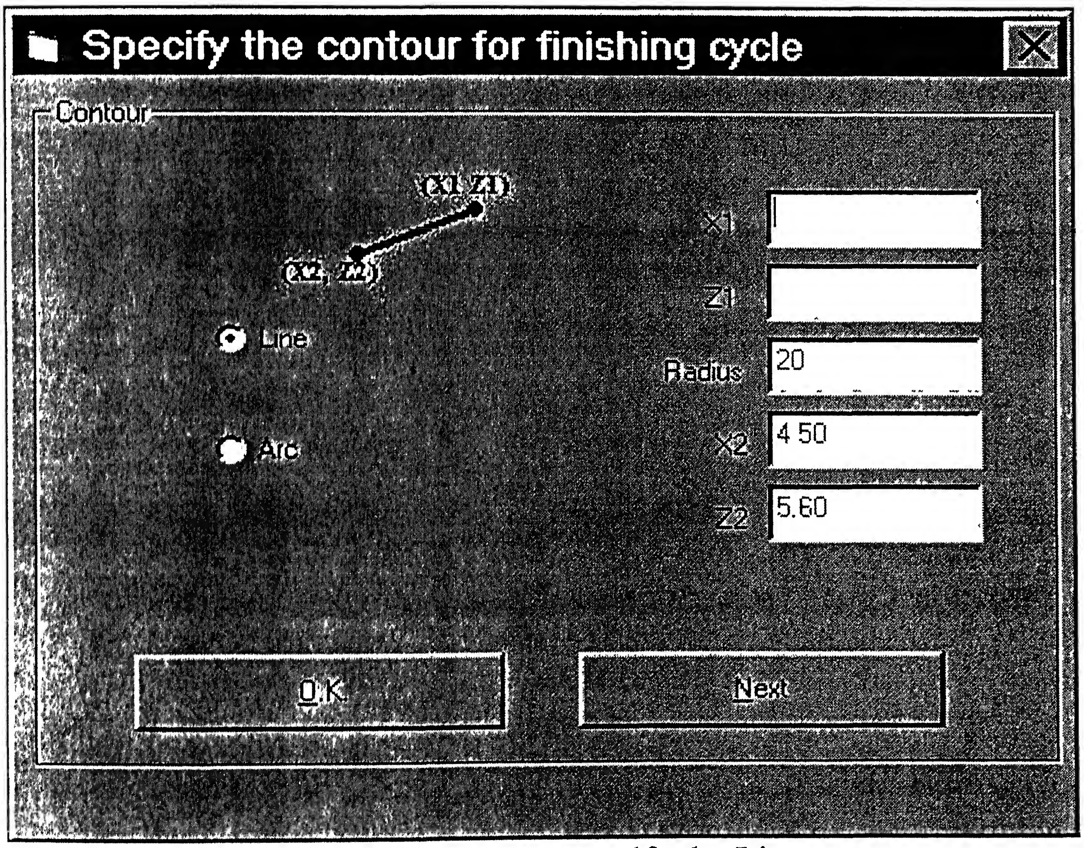


Figure 3.9. Specify the Line

3.5. Process sequencing approach in the proposed software

The process sequencing for the rough turning operation has been done using G90, G94 cycle. In CNC lathe machine (0T FANUC series), there are two options available for the rough turning operations.

(1) The G71 cycle and

(2) G90 / 94 cycle.

3.5.1. G71 (Rough Turning Cycle)

The G71 command automatically generates roughing passes to turn a workpiece to a specified profile, leaving an allowance for finishing. It reads a program segment specified by the P and Q letter addresses and determines the number of passes, the depth of cut for each pass, and the number of repeat passes for the cycle.

Example G71

```
%
:1071
N5 G90 G20
N10 T0101
N15 M03
N20 G00 X2 Z0.1 M08
N35 G71 P40 Q55 U0.05 W0.05 D625 F0.012
N40 G01 X0 Z0
N45 G03 X1 Z-0.5 I0 K-0.5
N50 G01 Z-1.0
N55 X2.1 Z-1.5
N60 T0100 G00 X4 Z3
N65 T0202
N70 G00 X2 Z0.1
N75 G70 P40 Q55 F0.006
N80 G00 X4 Z3 M09
N85 T0200 M05
N90 M30
```

3.5.2. G90 / G94 Rough Turning Cycle

The G90 cycle is commonly called single pass roughing. This simple cycle will machine one complete rough pass. The tool will rapid to the cutting diameter, turn the part, retract on X and Z, and return to the starting position.

The G90 cycle is programmed as follows:

G90X2.0Z-5.0F.012

Where X2.0= The cutting diameter.

Where Z-5.0 = The end point in Z axis for the cut.

Where F.012 = The feed rate for the cut motion.

The program should have the tool placed at the convenient start position before programming the G90 cycle. The convenient start position is usually the rough stock diameter for X and the rough stock face for Z. One line of command causes 4 distinct motion of the tool.

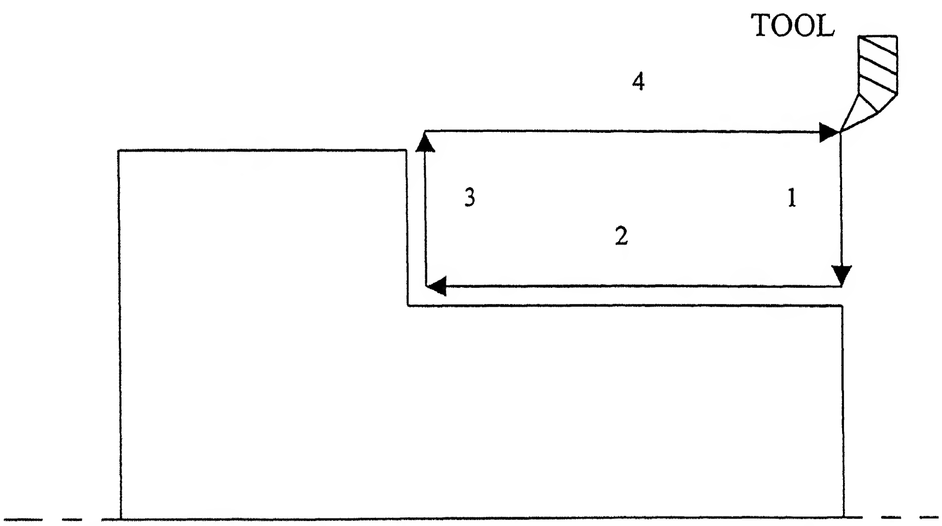


Figure 3.10. G90 cycle

G94 cycle (Like G90 it is also a single Box type Cycle) is essentially the same except that it is used when the major direction of cut is along X axis.

In the proposed software an approach has been investigated to closely monitor the sequence of the turning processes using distinct G90 / 94 cycles instead of one G71 Cycle.

Following findings have been observed during monitoring the roughing operation on CNC lathe.

- G71 cycle is useful from the point of view of simplicity. The user has to make the CNC part program by defining the contour and inputting the diameter and length of the workpiece. The rest of the things are done by the G71 cycle itself. Where as in case of G90 / 94 Cycle one has to go step by step. Firstly, one has to identify the volume of the material to be rough turned then the initial position of the tool is to be inputted.
- There is no provision for the facing operation in case of G71 cycle whereas in G94 cycle the provision for the facing operation exists. It can be proved to be particularly useful when the major direction of cut is along X axis instead of Z axis. This is illustrated below:

From the figure the box represents the material to be removed from a workpiece. If G71 cycle is applied to remove this material by turning it will take about 9 repetitions whereas by G94 facing operation it will be done in only 4 repetitions thus reducing the time taken for machining. In the process sequencing algorithm this has been taken into consideration. The algorithm decides whether to use G90 or G94 so that the machining time can be reduced.

- One more advantage of G90 / 94 cycle over G71 is the more closer monitoring of the process sequencing is possible. This can be particularly helpful for the slender workpieces where the length to diameter ratio is high. In this case the bending force on the work piece is considerable. So to avoid any workpiece failure due to bending it is necessary to do the turning operations from right to left as far as possible. The process sequencing algorithm also takes care of this fact.

The proposed iterative algorithm for process sequencing firstly compares the blank size with the extreme dimensions of the work piece. If it finds that the blank size is appropriate for obtaining the workpiece(Drawing of AutoCAD) it allows for the furthur process sequencing otherwise a message box appears on the screen alarming the user to change the workpiece.

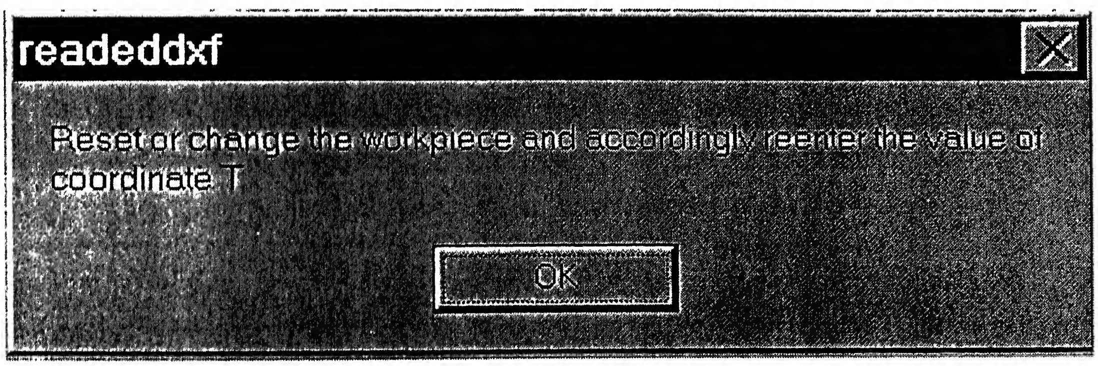


Figure 3.11. Message box

After this the algorithm takes decision about the volume of the material to be removed. It is done with the help of the files which are automatically created and saved while the standard features of the drawing. The iterative algorithm is as below:

The algorithm compares the coordinates of all the features with a reference point with coordinates (P along the X axis and Q along the Z axis). After each identification for the volume the reference piont (P, Q) is modified. After the modification again it compares all the coordinates of the feature with the modified value.

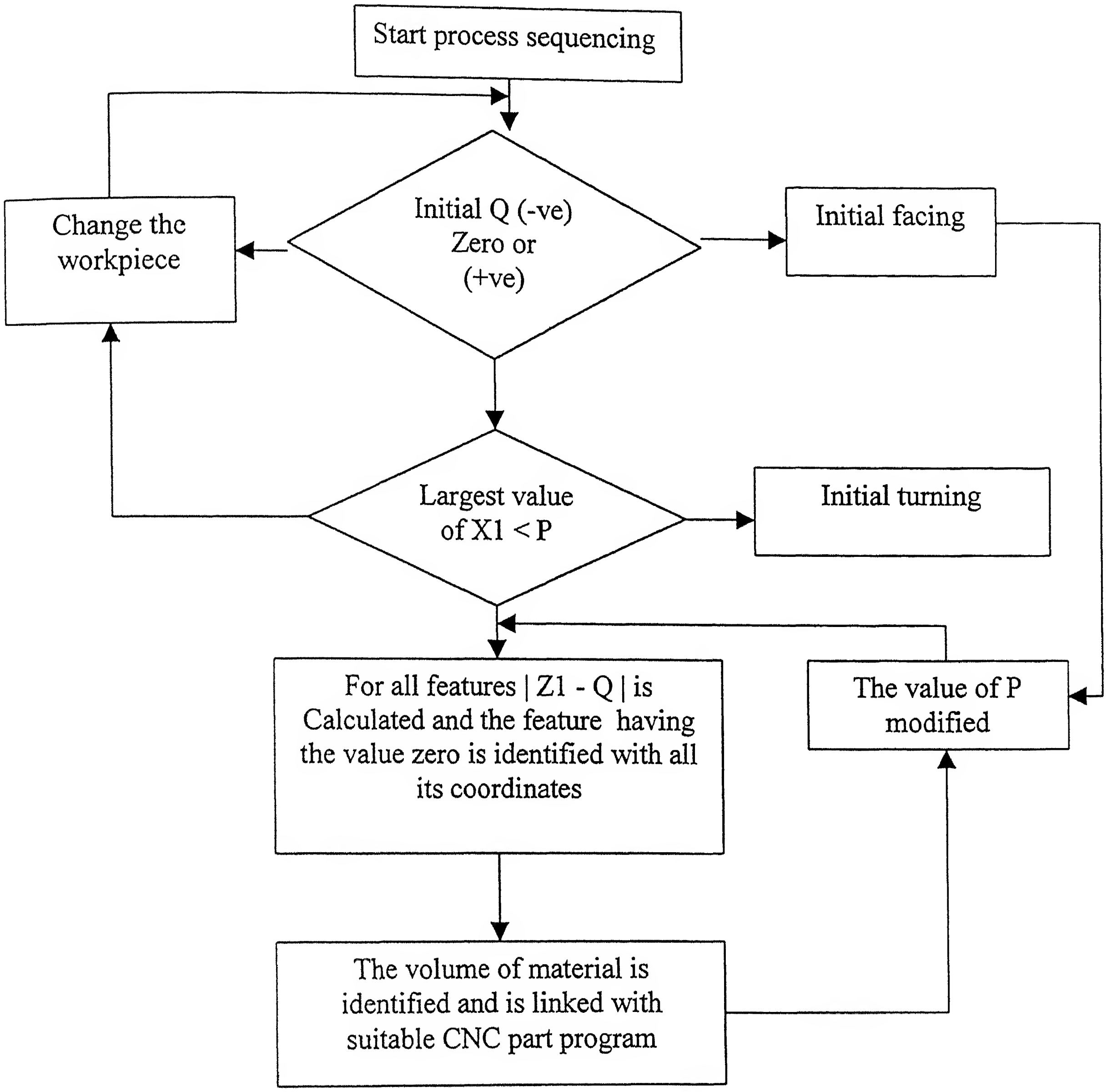
X1 = Value of the X coordinate of the feature.

Z1 = Value of the Y coordinate of the feature.

P = Value of the X coordinate of the reference point.

Q = Value of the Y coordinate of the reference point.

The iterative algorithm for process sequencing



For Rough Material Removal

If N is the total number of the features then Iteration takes place N number of times and after that exits from the loop automatically.

After the rough turning takes place the user is required to specify the contour for finishing then the special features are taken into consideration. Throughout the application user is assisted by sketches and pictures placed on the screen so that he may have a the feeling about what to do next. The software also takes care about the possible human errors which may take place during specifying the coordinates. Lastly the CNC part program can be retrieved from a file where it is stored.

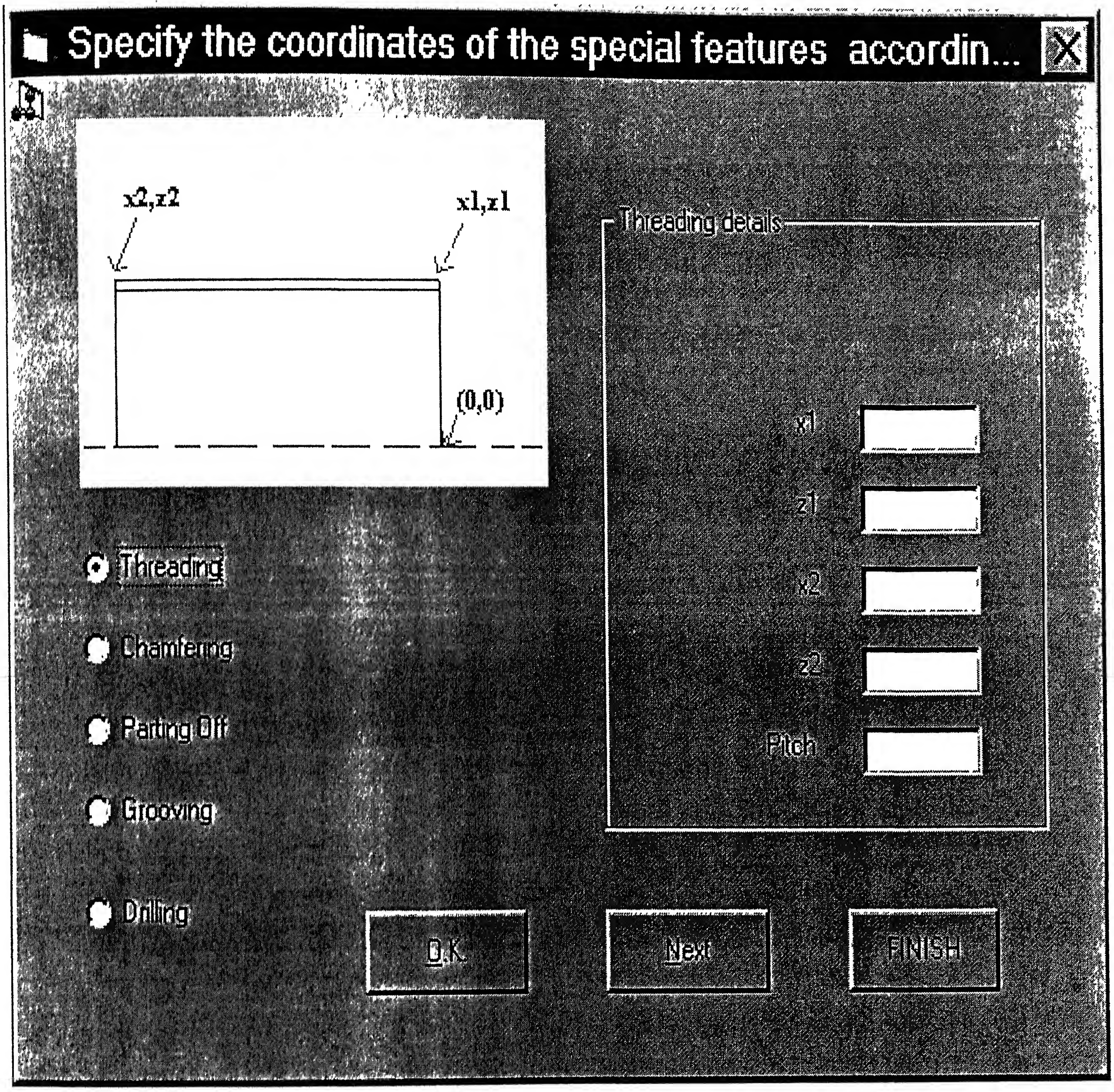


Figure 3.12. Specifying coordinates for Threading operation

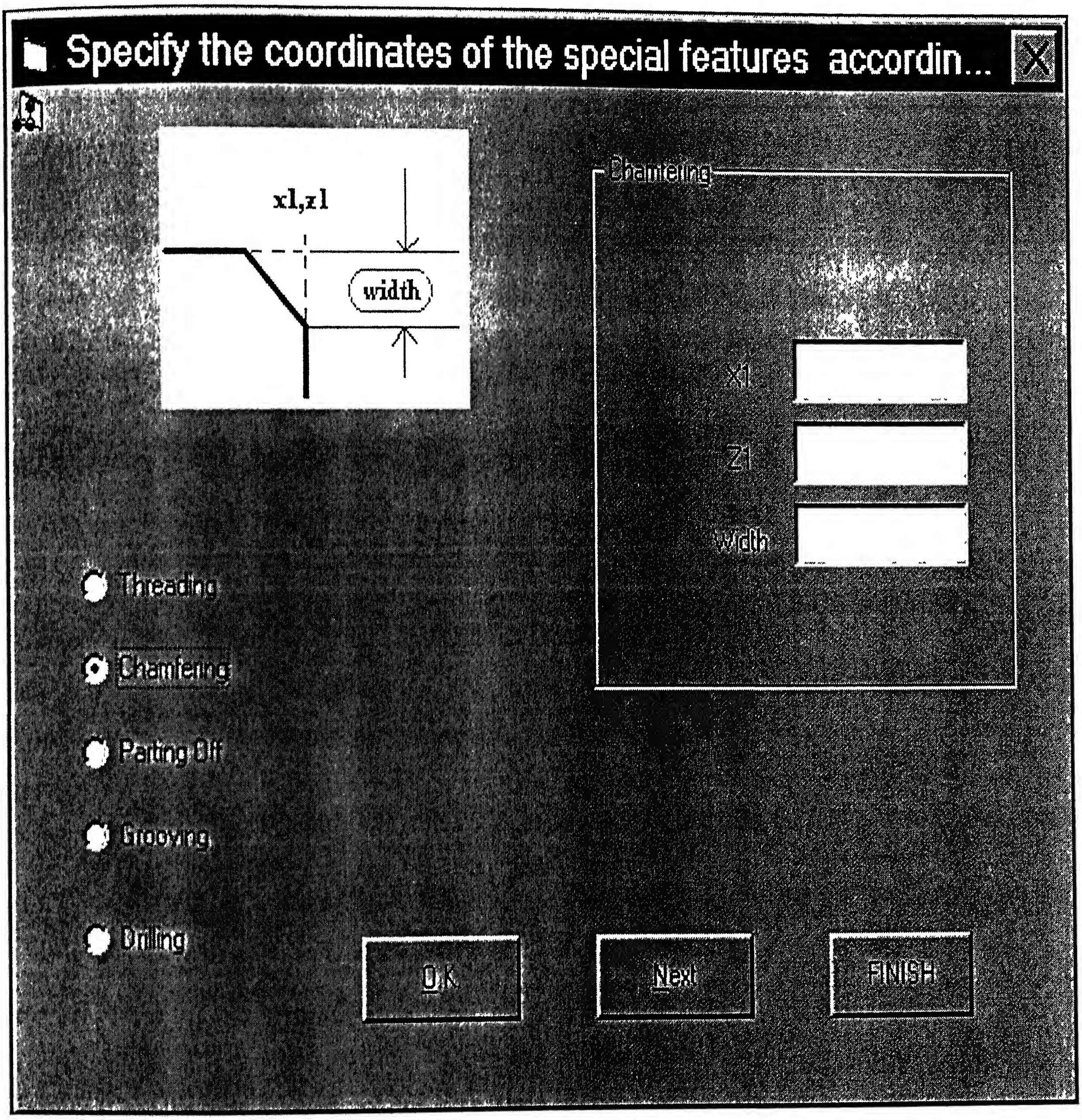


Figure 3.13. Specifying coordinates for Chamfering operation

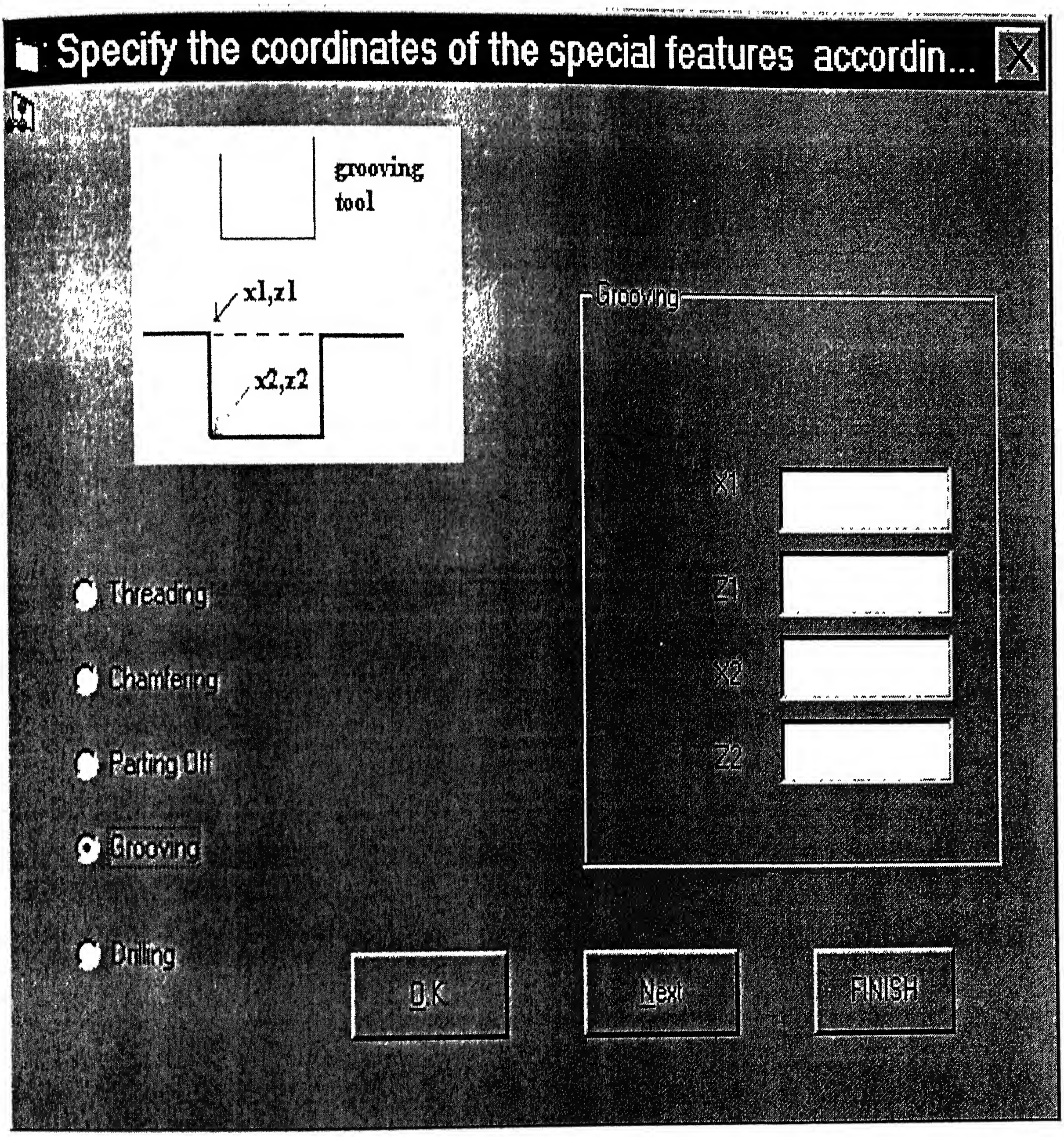


Figure 3.14. Specifying coordinates for Grooving operation

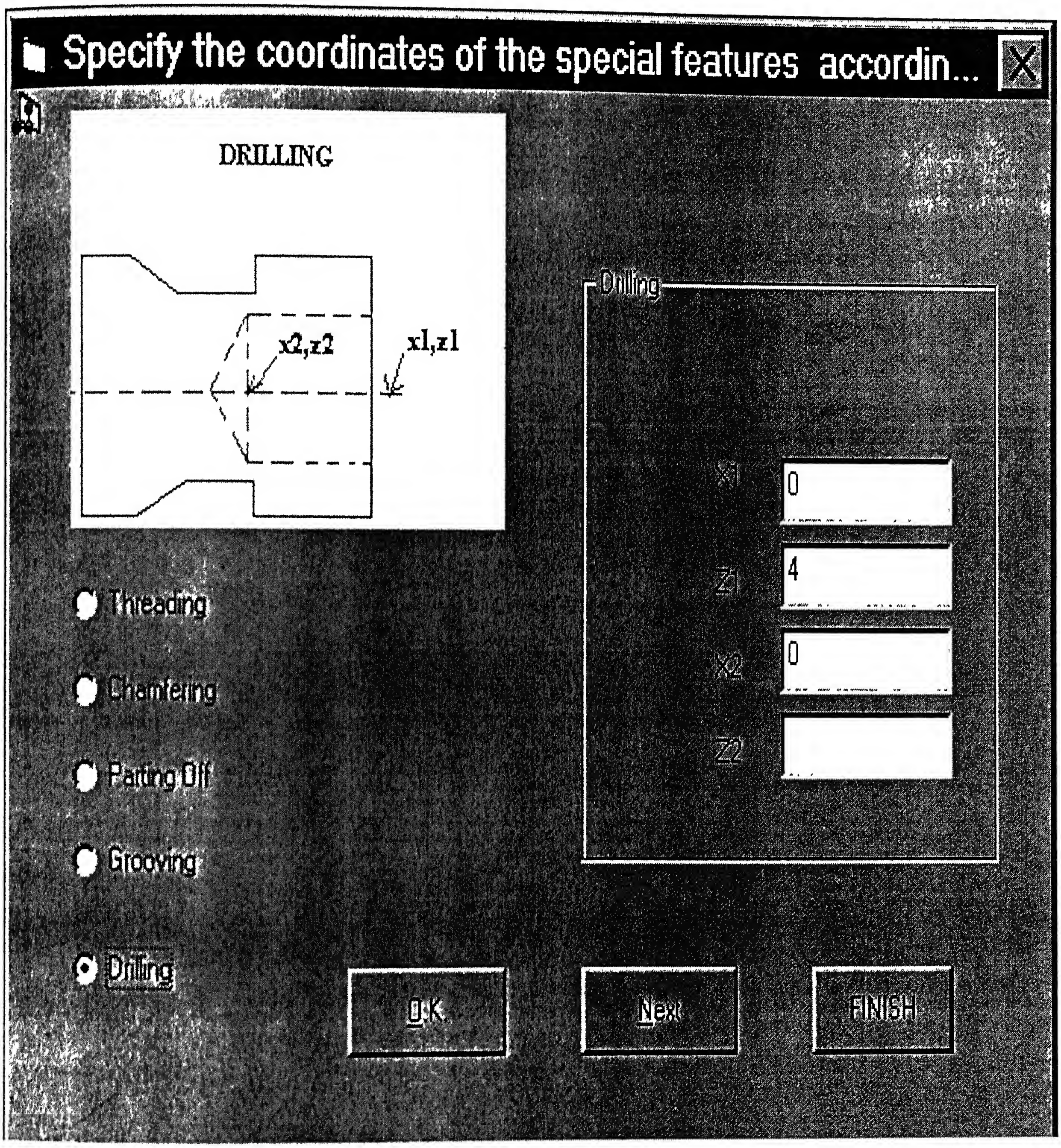


Figure 3.15. Specifying coordinates for Drilling operation

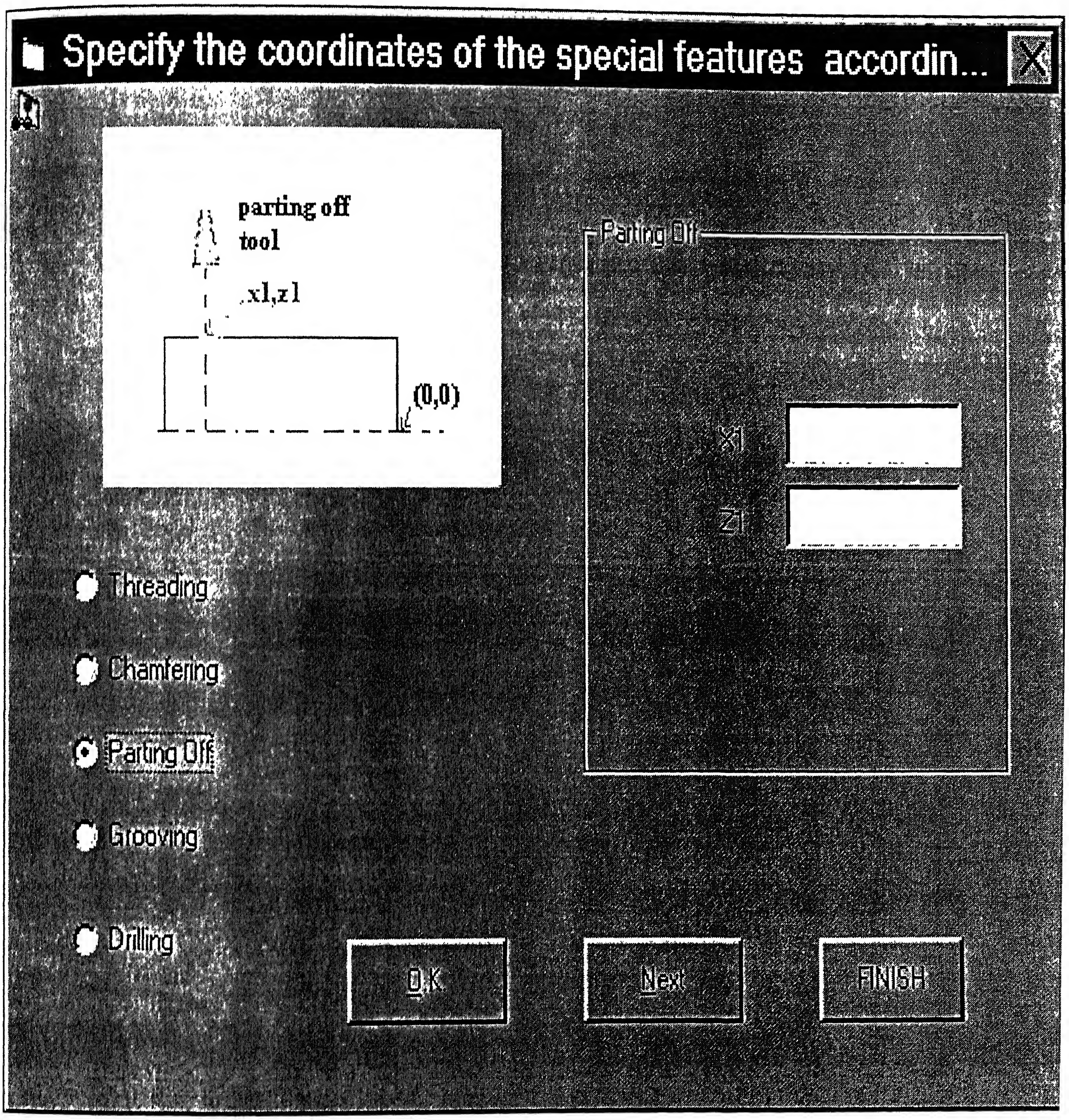


Figure 3.16. Specifying coordinates for Parting off operation

The CNC part program is generated automatically and stored in a separate file from which it can be retrieved and fed into the CNC turning machine. The software has been developed keeping in mind the user friendly aspect with top priority. The main aim was to integrate the AutoCAD software with Visual Basic so that the user don't have to take care about the various details of the coordinates. All computational works are done by the software itself.

The helps are available in the form of tool tip text throughout the span of running. The software is such that any person having some idea about AutoCAD can use it.

This completes the description of the software.

Chapter 4

Implementation and Results

Feature Recognition and Designing with Features have been recognised as alternative approaches to the integration of design and manufacturing functions. In this thesis the approach has been investigated by developing a feature-based computer aided part programming with AutoCAD. The geometric representation of the component in terms of manufacturing features in AutoCAD is captured in a form suitable for outlining process sequencing. The objective of the thesis is to develop software suitable for rotational or turned components and outlining process sequencing.

The proposed application software package for CNC part program generation has been developed using Visual Basic and AutoCAD softwares. Visual BASIC has been used to create the graphic user interface(GUI) and also to perform all computational works.

The user specifies the inputs in the Visual Basic environment. The CNC part program is generated and stored in a separate file from where it can be used for the purpose of actual CNC turning. The package runs on any IBM compatible PC-386 and onwards in Windows environment. It is essential that the system should be loaded with AutoCAD and Visual Basic software. Has also been done to create an executable file out of the software package.

4.1. Inputs to the system

1. User is required to make the drawing in AuotCAD. It is advisable to change the Universal coordinate system of AutoCAD first. It is for the convinience of the user because X and Z axes of the drawing should have the same direction as that of CNC machine. Then the user is required to save the drawing in **.dwg** file extention. Just after that the same drawing is saved in **.dxf** file extention.

2. User is required to start the software package. The available blank size is inputted to the software.

3. The material of the tool and the workpiece material is then selected from combo boxes.

4. For special features like threads, chamfers, grooves and other operations the coordinates should be inputted by viewing the AutoCAD drawing. The coordinates to be inputted is displayed on the monitor so that even a lay man can input the data.

The user is required to give the inputs in the they are asked. Efforts have been made to make the package user-friendly. Most of the inputs (viz., specifying part geometry, selecting work-piece material and tool piece material etc.) can be done using mouse clicks. Hot keys have also been provided for those who are more comfortable with the key board. On line helps have been provided at the critical places. Checks and error messages have been incorporated at most of the places.

4.2. Outputs of the system

1. The standard features of the drawing and their coordinates.
2. CNC part program for the given blank and the required shape.
3. Tool path animation for the roughing cycle.

4.3. System files in Visual Basic

In Visual Basic a project consists of a group of files. Each *form* is saved separately with .FRM extension and each *code* module is saved in the files with .MAK extension. Some of the controls used in the application has been listed below.

Name(Prefix)	Control	Description
pic	Picture Box	Provides a rectangular area in which graphics can be displayed
lbl	Label	Displys text that the user can not modify at run time
cmd	Command	Responds to a mouse click by the user to activate an events
txt	Text Box	Displays text that the user can modify at run time.
cbo	Combo Box	Displays list of data. The user can select a value either by typing in the edit area or by selecting from the list
timer	Timer	Timer controls respond to the passage of time.

		They are independent of the user, and one can program them to take actions at regular intervals.

Table 4.1. Controls used in the software

In Visual Basic application each form(screen) contains several controls e.g., label, text box, picture box, command button etc. Each control contains one or more procedures like mouse-click, mouse-double click, key up, key down, got focus, lost focus etc. Each procedure performs a definite action when invoked. The list given below shows the various form names, its controls, procedures and the action performed.

Form Name	Control Name	Procedures	Action performed
Inte1a.frm	Cmd1a1	Mouse_click	Hides inte1a.frm and shows input.frm (the main screen)
	Cmd1a2	Mouse_click	Exits from the program. Application stops running
Input.frm	Mdemo menu	Mouse_click	Shows demonstration screen
	Mopen menu	Mouse_click	Shows a standard common dialog box for browsing the drawing
	Mstat menu	Mouse_click	Shows the form containing the number of various features
	Mprocess menu	Mouse_click	Shows the screen having the combo boxes for the tool and workpiece material selection
	Mani	Mouse_click	Sketches the workpiece Inside the blank on the screen
	Mcont	Mouse_click	Shows the screen for inputting of the

			coordinates of contour
	Maddf	Mouse_click	Shows the screen for inputting the coordinates of the special features.

Table 4.2. Some of the controls of the software

The Visual Basic integrated development environment (IDE) consists of the following elements.

Menu Bar

Displays the commands programmer use to work with Visual Basic. Besides the standard File, Edit, View, Window, and Help menus, menus are provided to access functions specific to programming such as Project, Format, or Debug.

Context Menus

Contain shortcuts to frequently performed actions. To open a context menu,one has to click the right mouse button on the object used. The specific list of shortcuts available from context menus depends on the part of the environment where programmer clicks the right mouse button.

Toolbars

Provide quick access to commonly used commands in the programming environment. Programmer clicks a button on the toolbar once to carry out the action represented by that button. By default, the Standard toolbar is displayed when programmer starts Visual Basic. Additional toolbars for editing, form design, and debugging can be toggled on or off from the Toolbars command on the View menu.

Toolbox

Provides a set of tools that programmers use at design time to place controls on a form. In addition to the default toolbox layout, one can create his own custom layouts by selecting Add Tab from the context menu and adding controls to the resulting tab.

Properties Window

Lists the property settings for the selected form or control. A *property* is a characteristic of an object, such as size, caption, or color.

Object Browser

Lists objects available for use in the project and gives a quick way to navigate through the code. One can use the Object Browser to explore objects in Visual Basic and other applications, see what methods and properties are available for those objects, and paste code procedures into the application.

Form Designer

Serves as a window that programmer customize to design the interface of the application. Programmer adds controls, graphics, and pictures to a form to create the look he wants. Each form in the application has its own form designer window.

Code Editor Window

Serves as an editor for entering application code. A separate code editor window is created for each form or code module in the application.

Form Layout Window

The Form Layout window allows programmer to position the forms in the application using a small graphical representation of the screen.

4.4. The results

The actual CNC part program generated can be viewed by actually running the software. The roughing operations are almost done automatically whereas for the generation of the CNC part program for special features parametric programming approach has been attempted.

Chapter 5

Conclusions and Scope for Future Work

5.1. Conclusions

While developing the interactive computer aided CNC part program generation system for rotational components every care has been taken to make it user friendly so that a novice in the field of CNC part programming is expected to end up in getting the good CNC program. So with the growing popularity of CNC machines in the industrial sector, this package can be of great help.

The package is a unique blend of AutoCAD and Visual Basic languages. AutoCAD has been used to make the drawing whereas Visual Basic on its part does the efficient graphic user interfacing.

A unique method for process sequencing has been proposed. The action of the software can be clearly subdivided into two parts. The first is for rough material removal and the second is machining for special features.

5.2. Suggestions for the future work

Computer aided CNC part program generation system for rotational components is a new field. Expansions and modifications at every stage is possible. However, some suggestions are listed below:

The system capability is limited due to non availability of sufficient data. The system can be improved by incorporating more and more data in the workpiece material and tool material list. In fact, it can be linked with the database related to these two for the selection of speed feed and depth of cut.

(a) the tool nose radius compensation concept can be added to the present work.

(b) The code can be modified to read the geometric information of the work piece for CNC milling also. The present work is restricted only to turning of rotational components.

(c) To increase the flexibility of the software it should be made such as to recognise more and more G and M codes.

(d) Very preliminary kind of attempt has been taken to animate the tool path. It can be extended taking into account the real speed feed and depth of cut.

Appendix A

DXF files

To open a DXF file

1 From the File menu, Open is chosen.

2. In the Select File dialog box, under Files of Type, DXF (*.dxf) is selected.

3. Then one of the following is done:

- The wanted DXF file is found and selected.
- Under File Name, name of the DXF file is entered.

4. Open is clicked .

A new drawing is created, and the DXF file is imported into the drawing.

Structure of DXF file

Essentially a DXF file is composed of pairs of codes and associated values. The codes, known as group codes, indicate the type of value that follows. Group codes and the associated values define a specific aspect of an object or entity. The line immediately following the group code is the associated value. This value can be a a string, an integer, or a floating-point value, such as the X coordinate of a point. The lines following the second line of the group, if any, are determined by the group definition and the data associated with the group. Special group codes are used as file separators, such as markers for the beginning and end of sections, tables, and the end of the file itself. Entities, objects, classes, tables and table entries, and file separators are introduced with a 0 group code that is followed by a name describing the group. The maximum DXF file string length is 256 characters. If AutoCAD drawing contains strings that exceed this number, those strings are truncated during SAVE, SAVEAS, and WBLOCK. OPEN and INSERT fail if your DXF file contains strings that exceed this number.

Using these group code and value pairs, a DXF file is organized into sections, composed of records, which are composed of a group code and a data item. Each group code and value is on its own line in the DXF file. Each section starts with a group code 0 followed by the string, SECTION. This is followed by a group code 2 and a string

indicating the name of the section (for example, HEADER). Each section is composed of group codes and values that define its elements. A section ends with a 0 followed by the string ENDSEC. If one use the Select Objects option of the SAVE and SAVEAS commands, the resulting DXF file contains only the ENTITIES section and the EOF marker. The ENTITIES section contains only the objects one select for output. If one select an insert entity, the corresponding block definition is not included in the output file.

- **Various sections in DXF file:**

➢ **HEADER Section**: Contains general information about the drawing. It consists of an AutoCAD database version number and a number of system variables. Each parameter contains a variable name and it's associated value. It contains the settings of variables associated with the drawing. Each variable is specified by a 9-group code giving the variable's name, followed by groups that supply the variable's value. New DXF header variables: CELWEIGHT, CEPSNID, CEPSNTYPE, DIMADEC, DIMALTRND, DIMATFIT, DIMAZIN, DIMDSEP, DIMFAC, DIMLDRBLK, DIMLUNIT, DIMLWD, DIMLWE, DIMTMOVE, ENDCAPS, EXTNAMES, FINGERPRINTGUID, HYPERLINKBASE, INSUNITS, JOINSTYLE, LWDISPLAY, PSTYLEMODE, PSVPSCALE, PUCSBASE, PUCSORGBACK, PUCSORGBOTTOM, PUCSORGFRONT, PUCSORGLEFT, PUCSORGRIGHT, PUCSORGTOP, PUCSORTHOREF, PUCSORTHOVIEW, TDUCREATE, TDUUPDATE, UCSBASE, UCSORGBACK, UCSORGBOTTOM, UCSORGFRONT, UCSORGLEFT, UCSORGBACK, UCSORGTOP, UCSORTHOREF, UCSORTHOVIEW, VERSIONGUID, XEDIT.

The following is an example of the HEADER section of a DXF file:

```
  0
SECTION
  2
HEADER      Beginning of HEADER section
```

9
$<variable>
<group code>
<value> Repeats for each header variable

0
ENDSEC End of HEADER section

- **CLASSES Section**: The CLASSES section holds the information for application-defined classes whose instances appear in the BLOCKS, ENTITIES, and OBJECTS sections of the database. It is assumed that a class definition is permanently fixed in the class hierarchy. All fields are required.

The following is an example of the CLASSES section of a DXF file:

0
SECTION
2
CLASSES Beginning of CLASSES section
0
CLASS
1
<class dxf record>
2
<class name>
3
<app name>

<flag>
280

<flag>

281

<flag> Repeats for each entry

0

ENDSEC End of CLASSES section

- **TABLES Section**: The TABLES section contains several tables, each of which can contain a variable number of entries. These codes are also used by AutoLISP® and ObjectARX applications in entity definition lists.

Common Symbol Table Group Codes: New code 330

Common Group Codes for Symbol Table Entries: New code 330

BLOCK_RECORD codes : 340, 310, 1001, 1000, 1002, 1070.

DIMSTYLE: New codes : 148, 79, 179, 276, 277, 278, 279, 341, 342, 343, 344, 371, 372. LAYER codes : 290, 370, 390.

UCS codes : 79, 146, 346, 71, 13, 23, 33.

VIEW codes : 281, 72, 110, 120, 130, 111, 121, 131, 112, 122, 132, 79, 146, 345, 346.

VPORT code : 281, 65, 110, 120, 130, 111, 121, 131, 112, 122, 132, 79, 146, 345, 346.

The following is an example of the TABLES section of a DXF file.

0

SECTION

2

TABLES Beginning of TABLES section

0

TABLE

2

<table type>

5

<handle>

100

AcDbSymbolTable

70
<max. entries> Common table group codes,
repeats for each entry

0
<table type>
5
<handle>
100
AcDbSymbolTableRecord
.
. <data>
. Table entry data, repeats,
for each table record
0
ENDTAB End of table

0
ENDSEC End of TABLES section

- **BLOCKS Section**: The BLOCKS section contains an entry for each block reference in the drawing. Contains block definition and drawing entities that make up each block reference in the drawing.

BLOCK codes : 330, 4. Also, new model/paper space definitions.

The following is an example of the BLOCKS section of a DXF file:

0
SECTION
2
BLOCKS Beginning of BLOCKS section

0

BLOCK

5

<handle>

100

AcDbEntity

8

<layer>

100

AcDbBlockBegin

2

<block name>

70

<flag>

10

<X value>

20

<Y value>

30

<Z value>

3

<block name>

1

<xref path> Begins each block entry

(a block entity definition)

0

<entity type>

.

. <data>

One entry for each entity definition within the block

0

ENDBLK

5

<handle>

100

AcDbBlockEnd End of each block entry

(an endblk entity definition)

0

ENDSEC End of BLOCKS section

- **ENTITIES Section** : Contains the graphical objects (entities) in the drawing, including block references (insert entities). In this section virtually all graphical objects are expressed along with there coordinates.

Common Group Codes for Entities: New codes 330, 410, 92, 310

New entities: ARCALIGNEDTEXT, RTEXT, WIPEOUT.

ACAD_PROXY_ENTITY codes 95, 70.

DIMENSION: Common Dimension Group Codes: codes 71, 72, 41, 42.

MTEXT codes : 73, 44.

The following is an example of the ENTITIES section of a DXF file:

0

SECTION

2

ENTITIES Beginning of ENTITIES section

0

<entity type>

5

<handle>

330

<pointer to owner>

100

AcDbEntity

8

<layer>

100

AcDb<classname>

.

. <data>

. One entry for each entity definition

0

ENDSEC End of ENTITIES section

- **OBJECTS Section** : It contains the nongraphical objects in the drawing. All objects that are not entities or symbol table records or symbol tables are stored in this section. Examples of entries in the OBJECTS section are dictionaries that contain mline styles and groups. Objects are similar to entities, except that they have no graphical or geometric meaning. All objects that are not entities or symbol table records or symbol tables are stored in this section. This section represents a homogeneous heap of objects with topological ordering of objects by ownership, such that the owners always appear before the objects they own.

Common Group Codes for Objects: New code 330.

New objects: ACDBDICTIONARYWDFLT, ACDBPLACEHOLDER, LAYOUT, PLOTSETTINGS.

ACAD_PROXY_OBJECT codes 95, 70.

DICTIONARY codes 280, 281.

XRECORD code 280.

The following is an example of the OBJECTS section of a DXF file:

0
SECTION
2
OBJECTS Beginning of OBJECTS section

0
DICTIONARY
5
<handle>
100
AcDbDictionary Beginning of named object
dictionary (root dictionary
object)

3
<dictionary name>
350
<handle of child> Repeats for each entry

0
<object type>
.
. <data>
. Groups of object data

ENDSEC End of OBJECTS section

Appendix B

List of G codes for 0 T Fanuc series CNC machine:

G00 Rapid traverse mode

G01 Feed rate mode

G02 Circular interpolation clockwise

G03 Circular interpolation counterclockwise

G04 Timed pause in program

G20 Inch mode

G21 Millimeter mode

G22 Software stroke limit on

G23 Software stroke limit off

G28 Return to home position in called axis

G29 Return from reference point

G32 Plain threading cycle

G40 Tool nose compensation cancel

G41 Tool nose compensation left

G42 Tool nose compensation right

G50 Set 0,0 position and maximum spindle speed

G70 Canned cycle, Finishing cycle (optional)

G71 Canned cycle, OD roughing cycle (optional)

G72 Canned cycle, Face roughing cycle (optional)

G73 Canned cycle, Profiling cycle (optional)

G74 Canned cycle, Face grooving cycle (optional)

G75 Canned cycle, OD grooving cycle (optional)

G76 Canned cycle, Thread cutting cycle (optional)

G90 Cutting cycle A

G92 Thread cutting cycle

G94 Cutting cycle B

G96 Constant surface speed mode, G97 cancel

G97 Constant RPM mode, G96 cancel

G98 Feed spec. in inches per minute, G99 cancel

G99 Feed spec. in inches per revolution, G98 cancel

List of M codes for 0 T Fanuc series CNC machine:

M00 Absolute stop

M01 Optional stop

M02 End of program

M03 Start spindle clockwise

M04 Start spindle counterclockwise

M05 Spindle stop

M06 Auto tool change

M07 Collet air off, close collet

M08 Coolant on

M09 Coolant off

M10 Air on (Open automatic door)

M11 Air off (Close automatic door)

M30 End of program

M98 Call subroutine

M99 Subroutine return

References

1. Arthur D., Richard C., Programming for Numerical Control Machines, Vol.2, 1976, pp. 151-210.

2. Sormaz D., Khoshnevis B., Process Sequencing and Process Clustering in Process Planning, Journal of Intelligent Manufacturing Systems, Vol. 7, 1996, pp. 189-200.

3. Vant Eve A. H. ,Kals H.J.J., A Generative CAPPS for Part Manufacturing,Vol.25, pp.325-329, 1986

4. Thakkar G., Design and development of Database for Process Planning and NC part program generation for rotational components.

5. Omura G., Mastering AutoCAD (for windows 98),Vol.3, 1999

6. Robert Callori B., Omura G., AutoCAD - Instant Reference,Vol.4, 2000

7. Perry G., Visual Basic 6.0, Vol.2,1999

8. Wilson F.W. (Editor in chief), Harvey P.D. (Assistant Editor), Tool Engineer's Hand book, Second edition, Mc-Graw Hill book company, 1959, pp.-1127-1275.

9. Bhattacharya A., Chandra sen G., Principles of machine tools, New central book Agency, Vol.2, 1975, pp.104-210.

10. Sandy A.H., A.M.I.P.E., Cutting tools for Engineers, Crosby lockwood and sons Ltd., 1991

11. Arthur W.J., Machine tools and operations, The caxton publishing company Ltd., 1985, Vol.5, pp. 329-335, 391-405.

12. Charles B.H., Numerical control for machine tools, Mc Graw Hill book company, 1972, Vol.5, pp. 123-144

13. Koenigsberger F., Design principles of metal cutting and machine tools, Pergamon press, 1964, pp. 223-324

14. Wagener A.M., Harlan R.A., Machine shop - Theory and Practice, Affiliated East and West Press Pvt. Ltd., 1976,pp. 1-70, 212-243.

15. American Society for Mechanical Engineers Research Committee on metal cutting data and bibliography, Manual on cutting of metals, 2nd edition , 1952

16. Peter S., A Comprehensive Guide to CNC Programming, first edition, 2000

17. http://www.hw.ac.uk/mecWWW/amu/amu.htm

18. http://www.asme.org/pubs/journals/online/

19. http://www.austinautocad.org/

20. http://www.autocode.com/getac.htm

21. http://www.axisnc.com/

22. http://www.bu.edu/library/sel/news.htm

23. http://asrp.alphalink.com.au/Cad/cad.html

24. http://itll.colorado.edu/GEEN1400/CADHelp/

25. http.//www.ieee.org/web/developers/webthes/index_c.htm

26. http://www.vickers.de/cae.htm

27. http://www.wokingham.demon.co.uk/prog/programming.htm

28. http://www.cncezpro com/gcodes.cfm

29. http://fstroj.utc.sk/journal/engl/018/018.htm

30. http://www.cuttingtoolengineering.com/

31. http://www.cmpsystems.com/default.htm

32. http://sciweb.lib umn.edu/subject/ej/mech.html

33. http //home earthlink.net/~proglog/gcode.htm

34. http://www.gefanuc.com/products/cnc/category.asp?cnc_id=56

35. http://www graytechsoftware com/products/gtworks/prod01-3d.html

36. http://www.bozdoc.f2s.com/CAD-History.htm

37. http://www.ijcc.org/Submission.html

38. http://www.fbe.unsw.edu.au/Learning/AutoCAD/CADnotes/

39. http://www.mfg.mtu.edu/cyberman/machtool/machtool/intro.html

40. http://ojps.aip.org/ASMEJournals/Manufacturing/?jsessionid=563911013158924064

41. http://www.ie.psu.edu/labs/mshop/lab1_7/lathe htm

42. http://www.uwstout.edu/mevl/mevl1.html

43. http://www.mts-cnc.com/incad.htm

44. http·//www.sciencedirect.com/science

45. http://www.researchpaper.com/

46. http://www.zarr.net/vb/download/default asp